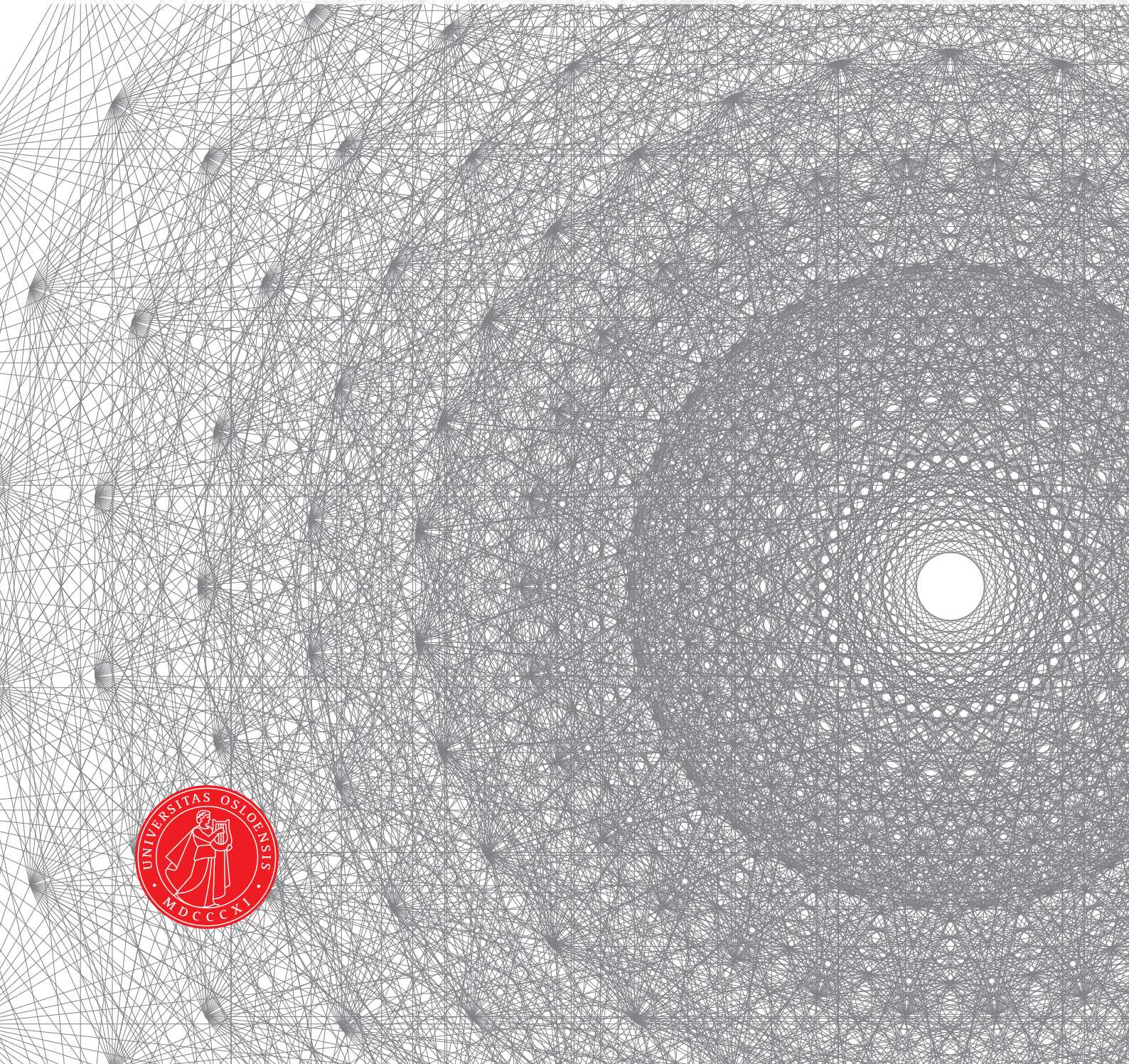
UiO : Department of Mathematics
University of Oslo

Analyzing and Predicting Demographics of NRK's Digital Users

Jenine Corrales

Master's Thesis, Spring 2019



This master's thesis is submitted under the master's program *Modelling and Data Analysis*, with program option *Statistics and Data Analysis*, at the Department of Mathematics, University of Oslo. The scope of the thesis is 60 credits.

The front page depicts a section of the root system of the exceptional Lie group E_8, projected into the plane. Lie groups were invented by the Norwegian mathematician Sophus Lie (1842–1899) to express symmetries in differential equations and today they play a central role in various parts of mathematics.

Abstract

The main objective of this thesis is to analyze the demographics of NRK's digital logged-in users, for which consumption behaviour data is also available. In particular, we examine NRK's reach across demographic groups by comparing the logged-in user population to the Norwegian population at large. In addition, we investigate the extent to which user demographics can be predicted based on users' digital content consumption behaviour. This is addressed by building classification models using known information on users and subsequently predicting on test sets, where results are then used to evaluate classifier performance. We examine in detail the quality of predictions made across classes as well as seek to determine whether or not these improve with quantity of content consumed.

Being able to predict user traits, such as gender and age, implies that there is some understanding of viewing patterns across demographic groups. For NRK this could mean for example, being able to identify and analyze variation in consumption within the population beyond a broad perspective.

We find that NRK has the most room for improvement in terms of reach amongst youth. We show that while age classification is challenging in a 6-class setting, improvements can be made by using instead 4 classes, where we can outperform the baseline by 15.2%. For gender classification we show that we can outperform the baseline by 17.3%. We also find that prediction accuracy has the tendency to increase with the quantity of unique contents consumed, for both age group and gender prediction.

Acknowledgements

I would first like to give my sincere thanks to my supervisors, Ida Scheel and Linn Cecilie Solbergersen. From them, I have received abundant guidance and feedback – all of which have been invaluable in the process of writing this thesis. I am especially grateful for my main supervisor, Ida, who has provided me with meticulous help every step of the way and has kept me in the right direction. My warmest gratitude to Linn Cecilie, whose insight and ideas were essential to my understanding. Thank you both for your patience and time, which I felt was always willingly given.

Thank you to NRK; the study of course could not have been possible without the granted access to such engaging data. The opportunity has been an amazing learning experience, and this I value greatly.

I would also like to thank my sisters, Jeanne and Kristin, as well as my friends, for the laughs and encouragement throughout the process. I appreciate you all for supplying me with fuel that I could not have otherwise gotten from caffeine.

And lastly, I would like to express my profound gratitude for my parents and the unconditional support they have always shown me.

Blindern, May 2019
Jenine Corrales

Contents

CHAPTER 1

Introduction

With the rise of digital technology there has been an emergence for the need to understand digital consumers and the material they consume. This is particularly important for businesses and service providers who aim to facilitate growth. Consuming products online such as music on Spotify, reading content on social media, and even buying physical merchandise from Amazon allows providers to gather information for potential use. This presents the valuable opportunity to extract knowledge and gain insight on a wide array of topics.

A customer-product understanding based on evidence can support important decision-making. For instance, the decision of what new product to develop may hinge upon knowing who the relevant target group is. For some, this may mean identifying a consumer group who has not yet been properly reached. Collected data in such a scenario can be analyzed and interpreted to aid in the company decision [15].

In another example, the wide use of social media has expanded the domain of personality prediction [17, 22, 28]. Personality traits have been shown to be indicative of retail relationships [31]. In this way, the ability to interpret and understand personality information can be particularly useful for advertisers aiming to make quality recommendations for aiding customers in identifying their needs or requirements. Such uses of consumer data demonstrate the importance of analyzing available information, and the broader impacts doing so might have e.g. on markets and society.

This thesis aims to analyze user data in order to gain a better understanding of consumer base. Two tasks we focus on is (1) assessing how user groups are distributed, and (2) predicting user information based on patterns of behaviour. Under this problem domain, we are primarily concerned with user demographics and consumption behaviour. For user demographics in particular, we segment users according to characteristics, such as age, gender, and geography, to obtain population groups for studying. Using information on these demographic groups (primarily population proportions and hence distribution) we form insight to aid in answering questions of particular interest (cf. Section 1.2). This addresses task (1). Task (2), on the other hand, is concerned with predicting the defined user demographics from user consumption behaviour in order to evaluate the ability to learn and extract patterns from our data.

Thus far, we have spoken about consumption in the broad sense of the word and its function as a link between customer and provider. To further expand on the notion for the context of this study, we refer to consumption behaviour as actions taken by consumers when using a product or service. For digital devices, this can entail accessing items, interacting with modules, mouse movements etc. The idea with using data related to consumption behaviour is that an individual's preferences underly their actions and therefore consumption behaviour is thought to reflect an aspect of a person's identity.

1.1 NRK

NRK, the Norwegian Public Broadcaster, is able to gather consumption behaviour data through the services they provide. Among these include, radio and TV services, as well as various types of content including news articles, opinion, culture, and lifestyle related content. These are made available via the internet across digital devices and platforms like web browsers and applications. When users access NRK and interact with content, information such as the type of content viewed and the time an event took place is logged. Since there is an abundant amount of information gathered, there is also a necessity to interpret the available material.

In addition to consumption data, NRK has a log-in service where users may optionally provide information on birth, gender, and postcode, i.e. demographic information. We refer to the users available on this service as *logged-in users.* The idea is to use both types of information (consumption and demographics) to aid in obtaining a deeper understanding of NRK's user base, like for example, determining what types of content different subgroups are interested in. Due to the availability of demographics, we therefore place our primary focus on the subset of users who are logged-in. In addition, the type of content consumption data analyzed is restricted to the TV content accessible on devices through the online streaming service, NRK TV.

As a non-commercial entity, NRK has the obligation to produce and distribute content for the Norwegian population. Once obtained, they can build upon an awareness of details about user demographics and its synergy with user behaviour to the benefit of this obligation. A key question that can be answered concerns where improvement in distributing their content might occur. To answer this, knowledge on which demographic groups NRK is reaching is necessary. Once this is known, the user group which is not so well reached can be identified, and thereby revealing where improvement can be made. This idea underlies one of the main subtopics studied in this thesis.

In having a substantial amount of consumption data, it may be useful for NRK to determine whether or not behaviour information can be used for prediction purposes. In particular, they would like to know if content consumption behaviour can be used to predict their users' demographics. An idea is that being able to do so successfully for logged-in users (i.e. those we have demographic data on) may mean also being able to do so for non-logged-in users. The ability to predict the demographics for the whole user base (as

opposed to simply logged-in users) may imply gaining more knowledge about user consumption, such as content interests for certain groups in the population. Having the ability to effectively analyze variations in viewership in such a way serves as a motivating factor for this thesis.

1.2 Research Questions

Having presented where the potential lies in analyzing NRK's data, we now introduce the scope of our research study. In particular, the main objective of this thesis is to perform a demographic analysis of NRK's logged-in users. We accomplish this by seeking to answer three main questions:

Q1: Which parts and to what degree does NRK reach various demographic groups of the Norwegian population, with respect to their logged-in user base?

Q2: To what extent can user demographics be predicted using information on content consumption behaviour?

Q3: Does the quality of predictions depend upon the quantity of consumed content?

1.3 Related Work

There exists research studies prior to this thesis that have also sought to predict user demographics based on user behaviour. Before proceeding further, we discuss previous papers related to our prediction task as we have defined it.

Thomas Krismayer et al. [19, 20] produced two similar papers on predicting user demographics from music listening information. In the first study the ability to substantially predict age, gender, and country is established – achieving a regression error 33.7% below the baseline error. In addition to logistic regression, other classifiers used for the research include support vector machines, decision trees, and naive bayes. The study additionally finds that an increase in listening events corresponds with an increase in classifier performance. The second paper acts as an extension to the first by considering the same problem domain but in addition, discovering that a similarity measure for the response can account for error in predictions. In addition, they find that the user information that can be derived from listening history can also help make better recommendations.

Different from media consumption, Hu et al.[12] describes a demographic prediction setting in which web browsing behaviour is used to predict gender and age. The results from modelling using support vector machines improve on baseline performance by 30.4% and 50.3% for gender and age prediction respectively. Kosinski et al. [17] uses user behaviour in the form of Facebook Likes to predict private traits and attributes. In this study logistic and linear regression are both used for predicting traits. The model used in one example, could distinguish between Democrats and Republicans with an accuracy of 85%. This study in particular discusses the implications of such predictive ability on

privacy.

There exists numerous other works which investigate problem domains similar to ours and the ones previously listed. Our contributions with this thesis include the following. Firstly, we consider the quality of predictions across individual age groups and models by using recall, precision, and F_1-score. Secondly, we examine the effects of grouping age intervals differently and how this might affect predictive ability. Thirdly, we use summary features derived from contextual data, such as user type, mode of viewing, as well as factor variables produced through Collaborative Filtering to supply information. Fourth, we observe the performance of the more simplistic KNN model against a more complex algorithm such as random forest. Finally, we determine if regularization improves prediction accuracy.

1.4 Outline of the Thesis

The remainder of this thesis is organized as follows. Chapter 2 presents the methodology which lays the foundation for the prediction task. This includes a discussion on classifiers, cross-validation, resampling and performance metrics. Chapter 3 consists of an exploration of the datasets involved in performing experiments and analyses. These consist of the previously described information on demographics and content consumption behaviour. In Chapter 4 the analyses are performed and reported. The final chapter, Chapter 5, discusses the research study and the overall findings.

CHAPTER 2

Methodology

This chapter provides the theoretical foundation underlying the prediction methodology we apply to our problem domain. We begin by presenting the relevant classifiers for this thesis and the learning mechanisms behind them. We then proceed with topics regarding model and data selection, such as cross-validation and resampling. This is then followed by a presentation of the performance metrics used to evaluate our classification results.

2.1 Classification

A main focus of this thesis is on the task of *supervised learning*, where for N observations in a dataset, the mapping between a known outcome measure, Y, and p input variables, $\boldsymbol{X} = (X_1, \ldots, X_p)$, is *learned* or approximated using an algorithm. That is, we seek to approximate f in,

$$Y = f(\boldsymbol{X}).$$

Ultimately, the goal is to apply f on observations which are not used in learning to obtain predictions, $\hat{Y}$, as accurately as possible. For this thesis, our target outcomes (age group and gender) are of qualitative nature and hence Y is characterized by K categories or classes. We therefore further define our task as a *classification problem*, i.e. defining a prediction rule f which categorizes observations into a group. In order to determine a particular prediction rule, we use observed values $(\boldsymbol{x}, y)$ to learn from. This set of observations is referred to as *training data*. The set of observations used to then test the accuracy of the learned prediction rule is called *test data*. We now proceed with the algorithms used in this study for learning from training data.

2.1.1 Multinomial Logistic Regression

The first classification method we describe is *multinomial logistic regression*. Multinomial logistic regression is a linear method which models the probability that an observation belongs to a particular class k provided linear functions of $\boldsymbol{x}$. Objects are classified into the class with the highest probability obtained by the model. For the special case in which $K = 2$, *binary logistic regression*, the object is classified into the class with probability greater than 0.5. Explicitly,

the model is defined as,

$$\begin{aligned} \log \frac{Pr(Y=1|\boldsymbol{x})}{Pr(Y=K|\boldsymbol{x})} &= \boldsymbol{\beta}_1 \cdot \boldsymbol{x} \\ \log \frac{Pr(Y=2|\boldsymbol{x})}{Pr(Y=K|\boldsymbol{x})} &= \boldsymbol{\beta}_2 \cdot \boldsymbol{x} \\ &\vdots \\ \log \frac{Pr(Y=K-1|\boldsymbol{x})}{Pr(Y=K|\boldsymbol{x})} &= \boldsymbol{\beta}_{K-1} \cdot \boldsymbol{x}, \end{aligned} \tag{2.1}$$

where $\boldsymbol{\beta}_k$ is the $p+1$ parameter vector associated with outcome k and $\boldsymbol{x}$ is a $p+1$ vector consisting of p explanatory variables and a constant term. Here, all K probabilities sum to one and each of $K-1$ outcomes are expressed as logit transformations against a last, arbitrary pivot class K. By transforming (2.1) to obtain probability expressions, $Pr(Y=k|\boldsymbol{x})$, $k=1,...,K-1$, and using that the probabilities sum to one, we may first arrive at the probability of class K occurring. In particular,

$$\begin{aligned} Pr(Y=K|\boldsymbol{x}) &= 1 - \sum_{k=1}^{K-1} Pr(Y=k|\boldsymbol{x}) \\ &= 1 - \sum_{k=1}^{K-1} Pr(Y=K|\boldsymbol{x}) \exp^{\boldsymbol{\beta}_k \cdot \boldsymbol{x}}, \end{aligned}$$

which gives,

$$\begin{aligned} Pr(Y=K|\boldsymbol{x}) + \sum_{k=1}^{K-1} Pr(Y=K|\boldsymbol{x}) \exp^{\boldsymbol{\beta}_k \cdot \boldsymbol{x}} &= 1 \\ Pr(Y=K|\boldsymbol{x}) \left\{ 1 + \sum_{k=1}^{K-1} \exp^{\boldsymbol{\beta}_k \cdot \boldsymbol{x}} \right\} &= 1. \end{aligned}$$

Hence,

$$Pr(Y=K|\boldsymbol{x}) = \frac{1}{1 + \sum_{k=1}^{K-1} \exp^{\boldsymbol{\beta}_k \cdot \boldsymbol{x}}}. \tag{2.2}$$

Then using this result we subsequently arrive at the probabilities for $k = 1,...,K-1$:

$$Pr(Y=k|\boldsymbol{x}) = \frac{\exp^{\boldsymbol{\beta}_k \cdot \boldsymbol{x}}}{1 + \sum_{\ell=1}^{K-1} \exp^{\boldsymbol{\beta}_\ell \cdot \boldsymbol{x}}}, k = 1, \ldots, K-1 \tag{2.3}$$

To fit the model (2.2)-(2.3), the maximum likelihood method is typically used [10], where regression coefficient values maximizing the probability of observing a given dataset are computed. Using the multinomial distribution, for a dataset containing N observations, with k being the class of observation i, and class probabilities $p_k(\boldsymbol{x}_i; \boldsymbol{\beta})$, the log-likelihood is given by,

$$\ell(\boldsymbol{\beta}) = \sum_{i=1}^{N} \log p_{y_i}(\boldsymbol{x}_i; \boldsymbol{\beta}). \tag{2.4}$$

It is then maximized by finding the derivative, equating to zero, and solving for $\boldsymbol{\beta}$. This is accomplished via the *Newton-Raphson algorithm* which produces equations updating obtained β-values through *iteratively reweighted least squares* [10].

Regularized Multinomial Logistic Regression

Regularization seeks to reduce overfitting on training data by imposing penalty terms in model fitting. For regularized multinomial logistic regression, the idea is to shrink large coefficients towards zero and by doing so effectively minimizing noise captured during training. The goal in mind is to obtain a model that generalizes better on test observations not used during model training. This is accomplished by penalizing complexity through a penalty term. The inclusion of a penalty term results in an objective function of the following form:

$$\max_{\boldsymbol{\beta}} \{\ell(\boldsymbol{\beta}) - \lambda R(\boldsymbol{\beta})\}, \tag{2.5}$$

where R is a regularization term and λ is a complexity parameter controlling the amount of coefficient shrinkage. Stated in this form, we see that for $\lambda = 0$, the penalty term has no effect and fitting equates to ordinary multinomial logistic regression as in (2.1)-(2.4). In general, as λ increases the degree of shrinkage also increases so as to reduce model flexibility. The choice of λ, therefore, has implications on the quality of model-fit and is chosen accordingly. The most commonly used method to optimize the value of λ is cross-validation, later discussed in Section 2.2. In our application, we use cross-validation to search through a grid of λ-values producing the optimal solution.

Two common approaches [10] to the regularization term, R, are *ridge* and *lasso* regularization. In ridge regression the regularization term is defined by the L2 norm [11],

$$R(\boldsymbol{\beta}) = \sum_{k=1}^{K-1} ||\boldsymbol{\beta}_k||_2^2 = \sum_{k=1}^{K-1} \sum_{j=1}^{p} \beta_{kj}^2, \tag{2.6}$$

while for lasso regression the regularization term is given by the L1 norm [32],

$$R(\boldsymbol{\beta}) = \sum_{k=1}^{K-1} ||\boldsymbol{\beta}_k||_1 = \sum_{k=1}^{K-1} \sum_{j=1}^{p} |\beta_{kj}|. \tag{2.7}$$

In the case of lasso regression, the choice of sufficiently large λ will lead to particular coefficients equaling to zero, leading to a type of subset selection.

2.1.2 K-Nearest Neighbors

K-Nearest Neighbors (KNN) is a non-parametric method using variable similarities to define the prediction rule. The algorithm identifies the closest[1] training points to an observation in terms of input $\boldsymbol{X}$ and determines the majority response, j, within this group. The number of data points from which to

[1] Measured in Euclidean distance with standardized variables[10].

evaluate the majority is specified by K. For KNN, the prediction rule for an observation is then defined as,

$$\hat{Y}(\boldsymbol{x}) = \arg\max_{j} \frac{1}{K} \sum_{i:\boldsymbol{x}_i \in N_K(\boldsymbol{x})} I(y_i = j), \tag{2.8}$$

where N_K is the K-nearest neighbors of $\boldsymbol{x}$ in the training set.

Here the parameter choice of K determines the level of model flexibility. For example, $K = 1$ implies a highly flexible model, as observations are classified according to the single nearest point. Alternatively, a large K leads to a less flexible model as generalization is extended across more neighboring points. KNN results in N/K neighborhoods, where each is fitted with a specific majority class. This means that the effective degrees of freedom for a KNN model is given by N/K. As with, the multinomial regression regularization term, the choice of K is optimized through a cross-validated (cf. Section 2.2) grid search of possible K-values.

2.1.3 Random Forests

The *random forests* classification algorithm is a method that constructs a model consisting of a classification tree ensemble (or collection), where each tree represents a vote for the final output class. In order to further detail the mechanisms behind the algorithm, we continue this subsection by laying the foundation upon which random forests is built on, namely the aforementioned classification tree and a method known as bagging. After doing so we proceed with further information on the random forests method.

Classification Trees

Classification trees produces a model with a tree-like structure consisting of internal nodes, branches, and leaf nodes. For the set of input variables $\boldsymbol{X} = (X_1, ..., X_p)$, each internal node represents a variable, X_j, and a corresponding split point s for that variable. Since each node includes a split, they lead to regions or branches of the tree produced by the choice of split. A branch can either lead to a subtree containing further splits or it can lead to a leaf node. Leaf nodes represent regions of final classifications for an observation which has fully traversed the tree. An example of such a tree is displayed in Figure 2.1. The diagram represents in particular, a *binary tree*, where at each internal node a split leads to two separate regions. In the figure, a split involving variable X_1 at cut point s_1 leads to a partition of two regions: $\{\boldsymbol{X}|X_1 \leq s_1\}$ and $\{\boldsymbol{X}|X_1 > s_1\}$.

The produced regions can then be further partitioned by considering another variable and split point combination. This is illustrated in Figure 2.1 by the branches leading from the internal nodes at level two of the tree. The splitting procedure is performed recursively, until some stopping criterion is reached. This stopping criterion can be, for example, when a certain number of observations are in each node. The final classification of an observation is then the mode of the leaf defined by the region R_m, for which it ultimately falls into. This means that for an object i that falls into leaf node m, the classification is determined

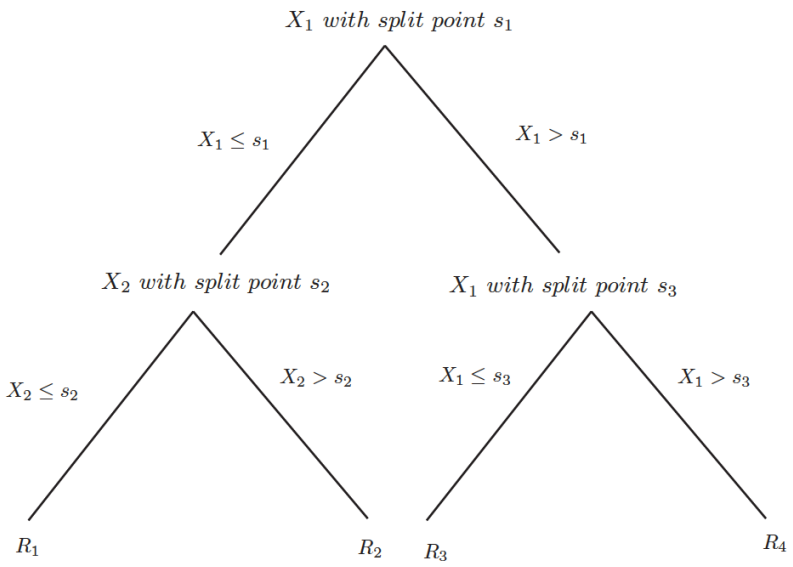


Figure 2.1: Classification tree with binary splits.

by the class k which satisfies $k(m) = \arg\max_k \hat{p}_{mk}$. Here $\hat{p}_{mk}$ denotes the class k proportions,

$$\hat{p}_{mk} = \frac{1}{N_m} \sum_{\boldsymbol{x}_i \in R_m} I(y_i = k),$$

and N_m denotes the number of observations in that node.

Intuitively, choosing meaningful splits in partitioning the input space leads to better model fit. Thus, the choice of variable X_j and cut point s at each internal node have direct implications on prediction quality. The task then at each node split is to satisfy some criterion, $Q_m(T)$, when selecting a combination of variable and cut point. This criterion can be thought of as a function minimizing the loss associated with the node split. For classification, the *Gini index* is a common criterion choice [2]:

$$\sum_{k \neq k'} \hat{p}_{mk}\hat{p}_{mk'} = \sum_{k=1}^{K} \hat{p}_{mk}(1 - \hat{p}_{mk}).$$

Typically, the structure of the tree is generated through an approach known as *recursive binary splitting*. It begins at the root node, where all observations belong to a single region, and proceeds by searching through all variable and

[2]Other options include:

- Misclassification error: $\frac{1}{N_m} \sum_{i \in R_m} I(y_i \neq k(m)) = 1 - \hat{p}_{mk(m)}$.
- Cross-entropy: $-\sum_{k=1}^{K} \hat{p}_{mk} \log \hat{p}_{mk}$.

However, in practice Gini index and cross-entropy are often preferred over misclassification error due to the applicability of numerical optimization.

cut point combinations to find the pair best satisfying the chosen criterion $Q_m(T)$. This is then successively performed, in a *greedy* manner – which means only determining the best split at that current step – for each resulting region further down the tree. The splitting ends when some stopping criterion is filled, e.g. when each leaf node has reached a certain size.

The classification trees method has the advantage of having the ability to capture complexities in training data, though for this reason it also often leads to noise and overfitting [10]. They thus have the tendency to generalize poorly on unseen test observations. To amend this, *tree pruning* is performed, where a large tree T_0 is fit to the data, followed by collapsing internal nodes in a process called *cost-complexity pruning*, which will be described shortly. When an internal node is collapsed, all branches coming from this node are elimnated resulting in a leaf node. The goal of cost-complexity pruning is to yield a subtree with improved test error rate.

To outline cost-complexity pruning we first let T denote a subtree of T_0 that has been pruned to have a corresponding number of leaf nodes $|T|$. The goal is to then minimize the cost function,

$$R_\alpha(T) = \sum_{m=1}^{|T|} N_m Q_m(T) + \alpha|T|, \tag{2.9}$$

for each α, in order to obtain subtree T_α. Breiman et al. (1984) detail that a unique T_α can indeed be found [4]. In (2.9), the first term consists of α, a regularization parameter controlling the tree's model fit and complexity, and $Q_m(T)$ reflecting error or accuracy. The cost function therefore expresses, for a subtree, error across leaf nodes and a penalty term. For $\alpha = 0$, the obtained tree is simply the entire tree T_0. As α increases, the size of T_α decreases. The choice of α, like other tuning parameters mentioned thus far, is produced by cross-validation.

Bagging

Bagging, which is short for *bootstrap aggregation*, is a technique that aggregates over a collection of B models, to obtain a classifier $\hat{Y}_{bag}(\boldsymbol{x})$ with reduced variance. From a given training set, B random samples, called *bootstrap samples*, are drawn with replacement and individually fit to form predictions $\hat{f}^{*b}(\boldsymbol{x}), b = 1, 2, ..., B$. In the case of classification trees, the procedure results in B bootstrap trees which differ in terms of chosen variables, number of nodes, and therefore differ in structure.

If we consider a classification problem with K classes, where $\hat{f}_{bag}(\boldsymbol{x})$ is a vector of length K containing the proportion of predictions to class k among the B trees, $[p_1(\boldsymbol{x}), p_2(\boldsymbol{x}), ..., p_K(\boldsymbol{x})]$, then the bagged classifier is defined as,

$$\hat{Y}_{bag}(\boldsymbol{x}) = \arg\max_k \hat{f}_{bag}(\boldsymbol{x}).$$

In words, an observation is predicted into the majority class for which it is classified among the collection of trees. The result is a reduction in variance for methods with high variance and hence increased stability [10].

Ensemble Learning in Random Forests

Random forests, like bagging constructs a classifier composed of an ensemble of bootstrapped trees. However, it improves on the bagging method by decorrelating the trees produced by the bootstrap samples [3]. This is achieved by randomly selecting $m \leq p$ candidate variables for splitting in the process of growing a tree, where p is the total number of input variables. For a given observation, each tree from the ensemble votes for an output class k, and the final classification is the mode class produced by the ensemble. For classification, the random forest predictor is thus:

$$\hat{Y}_{rf}^{B}(\boldsymbol{x}) = \text{majority}\{\hat{Y}_b(\boldsymbol{x})\}_1^B,$$

where $\hat{Y}_b(\boldsymbol{x})$ is the classification for tree b.

The main idea is that by randomly selecting $m \leq p$ variables, a shrinkage in correlation is achieved. A smaller m forces the tree algorithm to consider different subsets of predictors at each split, hence producing different tree structures in the ensemble. This then allows for aggregating over less correlated trees, thus leading to improved variance. In addition, the inventors recommend choosing $m = \lfloor\sqrt{p}\rfloor$, though this is typically treated as a tuning parameter determined using *out-of-bag* estimates[10]. Out-of-bag estimates are, for an observation, $(\boldsymbol{x_i}, y_i)$, the majority vote among bootstrap trees not using $(\boldsymbol{x_i}, y_i)$.

2.1.4 Baseline Predictor

A baseline predictor uses a simplistic method to obtain prediction results. In this thesis it will be used as a reference point for comparing classifier performance. Here, the baseline method identifies the most frequently occurring class and predicts all test observations as belonging to that class.

2.2 Cross-Validation

The method most commonly used for evaluating model performance and parameter tuning is cross-validation [10]. It performs estimation by using a learned method, $\hat{f}(\boldsymbol{X})$, for predicting on sample data independent of that used for training. Essentially, this is accomplished by partitioning the available data into a training set for model fitting and an independent set for testing prediction performance.

We distinguish between two methods, the holdout method and K-fold cross-validation. Both procedures create mutually exclusive subsets of data. However, they differ in that the former splits the dataset into two parts while the latter partitions into K folds.

Holdout Approach

The holdout approach is a special case of cross-validation that randomly splits the dataset into two subsets, one used for model fitting (training set) and the second for predicting on new observations (test/holdout set), those of which are not included in the training process. The amount of data allocated to each set

may vary, though a common choice is to designate 2/3 of the data for training and to hold out the remaining 1/3 for testing [16].

Since the holdout method produces prediction results that vary according to the splitting of training and test set (due to randomness), it is performed k times. The overall performance is then evaluated based on an average of all k runs, a long with standard deviation [16]. The holdout method is a candidate validation technique when large datasets are involved due to time and computational costs [35].

K-Fold Cross Validation

An approach that mimics the ideal scenario of training and testing multiple times over to obtain an overall average is K-fold cross-validation. The first step in this method involves dividing the data into K partitions or *folds*, of roughly equal size. Then for $k = 1, 2, ..., K$, a model is fit using $K - 1$ folds and validated on the remaining fold k, such that each fold is utilized once.

Since the choice of K controls partition sizes, it also determines the training and validation set sizes. An increasingly large K decreases validation set size while simultaneously increasing the training set. Since the number of folds affects the training and validation set, it also influences bias and variance in obtained estimates. More specifically, when K is large, we obtain approximately unbiased estimates compared to small K, due to a larger training set. However, this also implies that the fitted submodels are based on more correlated training sets (due to overlap in training points), and hence leading to higher variance.

The choice of K therefore depends on this trade-off and should be chosen accordingly. Conventional values of K are five or ten as they have been shown to do better in terms of model error and computation requirements [5, 16].

K-fold cross-validation is often used to tune hyperparameters [10]. Given a parameter λ, a grid search is performed over a range of λ values. This is done by estimating the cross-validation error corresponding to each λ, and selecting that with the smallest error. The final model, however, is evaluated on a test set that is not used during the selection of λ.

2.3 Class Imbalance and Resampling

A concern in the domain of classification that is known to cause suboptimal results is the issue of class imbalance [24]. Class imbalance is the event in which one or more classes have prominently more observations than others. This can, for example, be a scenario in which the minority-majority class ratio is 1:1000.

The presence of class imbalance in a dataset results in biased predictions towards the majority class since minority classes tend to be overpowered in the learning process [8]. This issue can arise for example because there exists some constraint in data collection, e.g. limitations on contacting certain groups for surveys and questionnaires. This is in contrast to the instances when the imbalance is a natural occurrence e.g. in fraud detection when there are normally

more non-fraudulent occurrences than fraudulent [25].

There exists several techniques to counter the effects of class imbalance. These include for example, cost-sensitive learning, algorithmic-specific adaptations, and resampling [24]. In this thesis we adopt resampling – the process of sampling from a given dataset in order to obtain balanced class distributions. This is achieved by respectively adding or removing observations belonging to the minority classes or majority classes. Common approaches include random undersampling and random oversampling.

Random undersampling involves randomly selecting and eliminating observations from the most frequently occurring classes until all class sizes are equal [14]. Conversely, random oversampling creates new instances of minority classes by randomly selecting observations to replicate. Another proposed solution is the SMOTE method, an algorithm that creates synthetic instances for the minority class [7].

Several other techniques [1] have been developed cleverly to use available information in the process of rebalancing. Nonetheless, this thesis uses random undersampling as it has the advantage of simplicity, not requiring extensive strategizing with respect to data handling and having minimal computational costs.

2.4 Performance Metrics

In order to evaluate the extent to which predictions can be made by a given classification model, we enlist performance metrics that provide measures for prediction quality. These allow us to distinguish between poor performance and good performance.

For the purpose of illustration, we begin with the simple two-class classification problem that can be generalized to multi-class problems. We consider the four possible outcomes in a two-class problem, namely: true positive (TP), false positive (FP), false negative (FN), and true negative (TN). True positive and true negative outcomes together make up the case in which the classifier makes correct predictions. In particular, *true positive* outcomes are correct classifications into the class for which the instance occurs (positive), while *true negatives* are those correctly predicted as non-occurring instances of the class (negative). Correspondingly, misclassifications can be subdivided into two outcomes: those predicted to be positive but are actually negative (*false positive*) and those predicted to be negative but are actually positive (*false negative*). This is summarized in the confusion matrix illustrated in Table 2.1.

In a more general setting, such as a three-class problem displayed in Table 2.2, we use the *one-versus-rest* approach. This approach defines observations belonging to a class as positives and the remaining observations as negatives [2]. Hence, the definitions of TP, FP, FN, and TN are relative to a given class i. Specifically, TP_i's are those correctly predicted into a class i, FP_i's are those predicted into class i but do not truly belong to class i, FN_i's are those truly

		Observed	
		Positive	Negative
Predicted	Positive	TP	FP
	Negative	FN	TN

Table 2.1: Confusion matrix for classification with two categories.

belonging to class i but not classified as such, and TN_i's are neither classified into i nor truly belong to i.

		Observed		
		A	B	C
Predicted	A	6	3	1
	B	0	9	2
	C	4	1	5

Table 2.2: Confusion matrix for classification with three categories.

In the following we describe standard metrics typically used to summarize values in a confusion matrix.

Accuracy For a K-class problem with N observations to be classified, accuracy specifies the proportion of correctly classified predictions out of all classifications made,

$$A = \frac{\sum_{i=1}^{K} TP_i}{N}.$$

Recall For class i, recall quantifies how many objects are detected out of the actual total in that class,

$$R_i = \frac{TP_i}{TP_i + FN_i}.$$

Precision Out of the total predicted into a class i, precision gives the fraction that are correct predictions,

$$P_i = \frac{TP_i}{TP_i + FP_i}.$$

This measures a classifiers ability to detect relevant instances.

F-score The F-score incorporates both precision and recall into one measure. Formally, it is the harmonic mean [33] between the two and is defined as,

$$F_{\beta i} = \frac{(1 + \beta^2) P_i R_i}{R_i + \beta^2 P_i}.$$

Here, β adjusts the weight of importance for recall and precision. When both are of equal importance then $\beta = 1$, while a larger β value indicates that recall is of greater importance than precision and a small β implies the opposite. For this thesis we use the F-score with $\beta = 1$, as we are not inclined to favour either recall or precision.

Precision, recall, and F-score provide a closer look at how effective and exact a classifier is in terms of its ability to make correct predictions at the class level. Overall measures for recall, precision, and F-score can be obtained in two ways. The first method is called *macro-averaging*, which involves averaging the obtained metric for all classes. For example, a macro-averaged recall, in a three class problem would be obtained by summing R_1, R_1, and R_3 then dividing by three. The second method, *micro-averaging*, sums over the TP_i's, FN_i's and FP_i's (depending on the relevant metric) for each individual class to obtain an overall measure. As a concrete example, a micro-averaged recall for the matrix in Table 2.2 is obtained by summing over all TP_i's and dividing by the sum of all FN_i's and TP_i's:

$$\begin{aligned} R &= \frac{TP_1 + TP_2 + TP_3}{TP_1 + TP_2 + TP_3 + FN_1 + FN_2 + FN_3} \\ &= \frac{6 + 9 + 5}{6 + 9 + 5 + 4 + 4 + 3} \approx 0.65. \end{aligned}$$

Micro-averaging therefore considers each classified observation per class and is thus useful for obtaining overall recall, precision, and F-score in a class-imbalanced scenario. The micro-averaged recall, precision, and F-score, however, produce identical metric values as accuracy. We therefore only report one overall measure, which we later refer to as 'overall accuracy' in Chapter 4.

CHAPTER 3

Dataset

In this chapter, a description along with an exploration of the NRK datasets used during experimentation and analysis are provided. First, in Section 3.1 we describe content consumption data that characterizes viewer behaviour and which will form the covariates for our prediction task. This is then followed by an exploration of data containing demographic information of logged-in users in Section 3.2.The demographic information provides the response variables that will be used for our prediction task. In addition, we use it for analyzing NRK's reach.

3.1 Content Consumption Data

When a logged-in user views an episode from a series, or just a simple standalone program, NRK logs information pertaining to that specific viewing event. In this way, NRK is able to accumulate consumer data that directly characterizes patterns in viewership. We use this section to describe such data and how it is used to form the covariates for our prediction task. This data provided for us was obtained in February 2018.

Before proceeding further in this section, we clarify that *content*, henceforth, refers to series or programs available through NRK's TV service. Furthermore, we refer to interacting with such content as *viewing events*. This section goes on to describe the type of information that is gathered on TV content consumption by NRK's system, which we use in the sections that follow. Section 3.1.1 provides an overview of contextual data for viewing events, such as the concrete time a program is viewed or on what type of device. In Section 3.1.2 and Section 3.1.3 we describe variables that aim to summarize an individual's content preferences, and finally, in Section 3.1.4 we describe how the tables are joined to form a final working dataset that will later be used as the covariate matrix, $\boldsymbol{X}$, in our prediction task.

3.1.1 Contextual Data

We call the collected data describing the context of each viewing event *contextual data*, as it provides information involving the context for individual viewing events that take place. Each distinct event has a corresponding user, with context given by timestamp and device used. This dataset consists of 6,598,548 observations, where each row pertains to a distinct event with no

missing values. There exists 35,402 distinct users which are uniquely identified by an ID (`userId`). These user ID's can also exist in other datasets (later described) and are therefore use for joining datasets (also later described).

The raw contextual data is used to derive two variables summarizing how a user consumes content. In this subsection we describe how the two variables are formed and explore the information they produce. The first variable characterizes user type by determining what devices events are performed on. The second characterizes a user's tendency of viewing time.

Defining User Type Based on Devices Used

The type of device (`deviceCategory`) used for a specific viewing is characterized by the type of platform NRK is accessed on for that event. This can be a desktop computer, a mobile device, a tablet computer, or a television set. These are then further identified as running on an Apple TV, a web application, an iOS application, or android application. Since, our prediction task is concerned with classifying one individual into a demographic group, we have chosen to omit entries corresponding to a television device (e.g. Apple TV). This is due to TV's often being used by multiple people (for example, in a multi-member household), and therefore, may generally provide misleading or invalid information regarding a user account.

From device information, we construct a categorical variable (`userType`) describing a user in terms of what type of device they have used to access NRK. In particular, we are interested in whether or not they are iOS, android, or web users. This is done by producing, for each user ID, a vector of applications previously used in viewing events. A user is labeled as a web user if they have neither iOS nor android entries. To define an iOS user, we scan through all entries determining if they have at least one iOS device logged and none for android. If this is true, they are labeled as an iOS user. The converse rule applies, if a user has at least one android device logged and none for iOS, then they are categorized under android. For the case in which a user has both android and iOS in its device vector, they are labeled under the category 'both'. We underline that if a user has used both a web application and an iOS/android device, we overrule web in favour of iOS or android as this distinction is thought to be most important. The user type variable therefore consists of the labels: 'ios', 'android', 'both', and 'web'.

Figure 3.1 displays the distribution of user types. We see that the majority of users fall into the iOS category, with approximately 14000 users, followed by android and web users with approximately 8000 and 4500, respectively. The smallest category is 'both' with size less than 2000 users.

Determining Mode of Viewing Time

For each viewing event, time context for when the event took place is also logged. This takes the form of a date variable (`date`) and two timestamps each specifying:

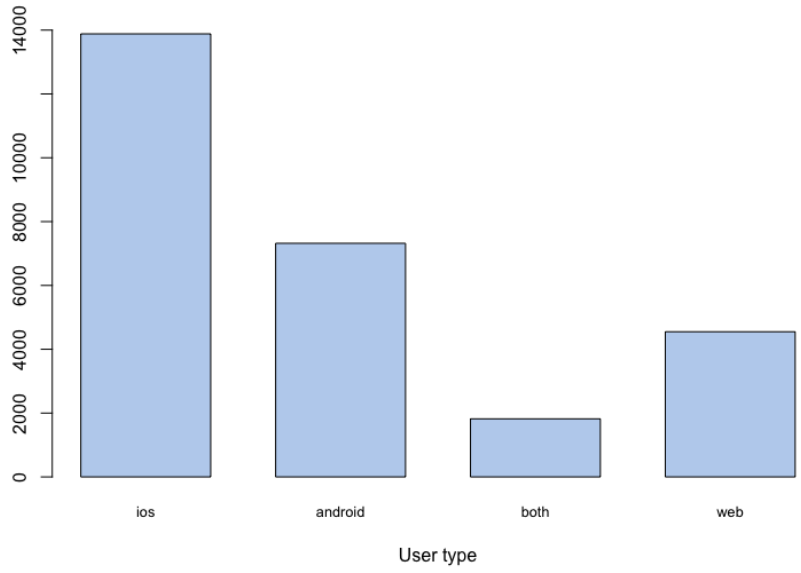


Figure 3.1: Distribution users in each user type category

(1) When a user began the event, i.e. began using the NRK platform (`visitStartTime`).

(2) The number of milliseconds after event start that the content is viewed (`timeOffset`).

Combined, they provide an exact timestamp for the start of a particular viewing event.

From date and time, we extract the particular day of the week and period of time in the day that an event occurred. This is then used to help us define a variable generalizing when a user most often views content. From the timestamp, we determine the hour of the day and categorize this as either, 'night', 'morning', 'daytime', 'afternoon', or 'evening'. The definitions of the categories are provided by NRK and are given as such:

Night: 03.00 - 06.00

Morning: 06.00 - 10.00

Daytime: 10.00 - 15.00

Afternoon: 15.00 - 19.00

Evening: 19.00 - 03.00.

The day of the week (Monday-Sunday) is then extracted from the given date and this is then categorized as either a 'weekday' (Monday-Friday) or 'weekend'

(Saturday-Sunday). The final time category (`timeCat`) corresponding to each event is then a combination of the day of the week and time of day for which the event occurs. For example, a viewing event that took place on a Monday at 12.00 is categorized as 'weekday + daytime'. The time category variable, thereby, has 10 possible categories: 'weekday + night', 'weekday + morning', 'weekday + daytime', 'weekday + afternoon', 'weekday + evening', 'weekend + night', 'weekend + morning', 'weekend + daytime', 'weekend + afternoon', 'weekend + evening'.

Finally, an event-time vector is constructed for each user, where an entry corresponds to the time category for which an event occurred. The mode of the entries in the vector is then found for each user, to provide the day and time combination that a user most often views content (`timeCat`). For the multimodal case a category, 'multiple', exists specifying that the user has two or more modes of viewing time.

Figure 3.2 displays how mode of viewing time is distributed across users. It reveals that the majority of users (approximately 14,000) most often consume content on weekday evenings. Conversely, the least amount of users consume content on weekend nights (and weekday nights), constituting only a fraction of users (fewer than 1,000 in each case). The second largest time category is weekend evenings, followed by weekday afternoons. This suggests that users tend to be more active later in the day compared to early mornings. Lastly, we see that a moderate amount fall into the category 'multiple'.

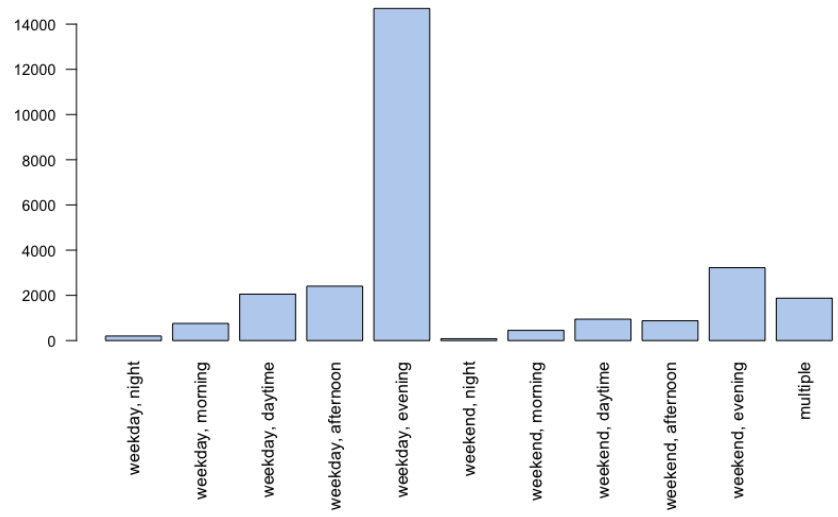


Figure 3.2: Distribution of user's viewing mode in terms of time of day and week.

3.1.2 Collaborative Filtering Factor Variables

Along with information on viewing context, NRK has provided a set of factor variables profiling the content preferences of logged-in users. The dataset consists of 193,625 rows of distinct logged-in users, one column with the number of unique contents seen by each user, and 20 numerical columns uniquely summarizing each individual's viewing preference patterns. We call these 20 columns *factor variables*. They describe interactions between users and items. These 20 columns define what we refer to as *factor variables*.

The factor variables are a product of a method called *Collaborative Filtering*. The goal of Collaborative Filtering is to predict items of potential interest for a user. It bases its predictions on collective user preferences. The premise is that a person is likely to choose an item that users with similar tastes prefer. To analyze the concept of taste, user preferences are represented in a matrix $\boldsymbol{A}$ of dimension $U \times I$, where rows represent users and columns represent items. The element in row u and column i hence expresses user u's taste for item i.

Preferences are expressed as feedback provided by the user. Conventionally, the feedback collected is either *explicit* or *implicit*. Explicit feedback constitutes direct reactions to items as expressed by the users. An example of such is the option to give a thumbs up or a thumbs down on a YouTube video. Implicit feedback consists of user actions such as, browsing history or the number of clicks on an item. In this way, a user reveals preference by how much they interact with an item, i.e. more interaction implies more interest. NRK specifically, uses implicit feedback based on how a user interacts with TV programs. In this case then, $\boldsymbol{A} \in 0, 1^{U \times I}$ is a binary 0-1 matrix where entry $a_{ui} = 1$ if a user has interacted with item i, and 0 otherwise. In addition, $\boldsymbol{R} \in \mathbb{R}^{R \times I}$ is a matrix where element r_{ui} quantifies how much interaction user u has had with item i.

The Collaborative Filtering approach that NRK uses decomposes the preference matrix $\boldsymbol{A}$ into two lower dimension matrices $\boldsymbol{Q} \in \mathbb{R}^{U \times f}$ and $\boldsymbol{W}^T \in \mathbb{R}^{f \times I}$ such that,

$$\boldsymbol{A} \approx \boldsymbol{Q}\boldsymbol{W}^T. \tag{3.1}$$

An important goal of this matrix factorization is dimension reduction by identifying necessary *latent factors*. The basic idea is that preference can be described by essential features, i.e. latent factors – as opposed to using the entire matrix $\boldsymbol{A}$. A user's preference for a program is rooted in how much they like its defining features, for example language, origin, etc. Thus in (3.1), $\boldsymbol{Q}$ is a user-factor matrix where each row specifies a user's interest in the set of latent factors, while the item-factor matrix $\boldsymbol{W}$ specifies each item's possession of those factors. If we consider user and item row vectors $\boldsymbol{q}_u \in \mathbb{R}^f$ and $\boldsymbol{w}_i \in \mathbb{R}^f$, from $\boldsymbol{Q}$ and $\boldsymbol{W}$, respectively, an interpretation of their inner product, $\boldsymbol{q}_u^T \boldsymbol{w}_i$, is then a measure of suitability between user and item based on latent factors. Once the latent factors are determined, $\boldsymbol{q}_u^T \boldsymbol{w}_i$ estimates a user-item preference entry, a_{ui}, in $\boldsymbol{A}$.

For the weighted regularized matrix factorization method of Hu et al. [13] that NRK uses, finding the latent factors involves minimizing some loss function

of the form,

$$\text{loss}(\boldsymbol{Q}, \boldsymbol{W}) = \sum_{u,i} c(r_{ui})(a_{ui} - \boldsymbol{q}_u^T \boldsymbol{w}_i)^2 + \lambda(||\boldsymbol{q}_u||^2 + ||\boldsymbol{w}_i||^2),$$

where $c(r_{ui}) = 1 + \alpha r_{ui}$ is a function that expresses the confidence we have in that a_{ui} expresses whether user u has a preference for item i or not, with α being a tuning parameter. Both α and λ, are regularization terms controlling for overfitting, are tuned via cross-validation.

NRK uses the Spark implementation [1] to identify the latent factors. For our classification task we use the obtained user-factor matrix $\boldsymbol{Q}$. Since the user-factor matrix reflects interest in certain item features and hence provides a description of content preferences, we hypothesize that they can be used to predict demographic groups.

In addition to the factor variables, the dataset acquired includes the quantity of unique contents viewed by each user (`uniqueContents`). This variable is not used as a covariate for model building, but instead used to group predictions by how much a user has seen (cf. Chapter 4). The quantities range from 0 to 13,748 unique contents viewed with a median of 40.

3.1.3 Content Genre Variables

NRK has also provided a dataset constituting of, what we refer to as, *content genre variables*. The dataset consists of 49,981 observations, each pertaining to a distinct user and it's corresponding content genre variables. The content genre variables are comprised of three columns, which together summarize the genre of content consumed by each individual. The variables are roughly such that `variable1` places the viewed content on a scale from entertaining to educational, `variable2` on a scale from emotional to fact-driven, and `variable3` on a scale from traditional to contemporary.

The three variables are constructed by manually giving each program or series a score {-2, -1, 1, 2} for variable 1 and variable 2; and {-1,0,1} for variable 3. The final scores for each user and variable is then obtained by taking the average score of that variable of all content viewed by the user. We note that this method in particular is developed by NRK and is thus far considered experimental.

3.1.4 Design Matrix

Using the content consumption datasets provided for us, we construct a design matrix consisting of the input variables to be used in our prediction task. The design matrix is produced by merging the relevant input variables described in Sections 3.1.1 to 3.1.3. These include, user type, viewing time mode, the 20 factor variables, and the content genre variables. The merging is performed such that the resulting dataset contains only the logged-in users with existing information across all input variables. Since these input variables uniquely

[1] https://spark.apache.org/docs/latest/mllib-collaborative-filtering.html

describe each user, all observations in the dataset are distinct.

The categorical input (user type and viewing time mode) are coded using dummy variables. More specifically, the labels corresponding to a categorical input are each converted to its own binary (1-0) variable except for one reference category. For example, from user type, we obtain three new binary variables: `ios`, `both`, and `web` (implying that android is the reference category). An iOS user will then have the value 1 registered under `ios` and 0 for the rest, while an android user will have 0 for all 3 variables. Viewing time mode is analogously coded with reference category set to 'multiple'.

Additionally, the design matrix is further reduced to the subset of logged-in users that exist on the demographic dataset (discussed in Section 3.2). This is due to the fact that the demographic dataset provides the response variables for our classification task. Ultimately, we are left with a dataset consisting of 27,571 observations, where each row represents a distinct user characterized by unique content consumption behaviour.

3.2 Demographic Data

We now proceed with describing the dataset containing NRK's demographic information on logged-in users. Like the content consumption data, the demographic data set was obtained in February 2018 and was gathered through NRK's log-in service where users have the option to create personal accounts. These are accessible across various digital devices such as mobile phones, tablets, desktop computers, and television sets. Through the log-in service, users are provided with the option to register the following information[2]: birthdate, gender, and postal code.

Initially, the gathered data consists of 62,943 observations, each corresponding to a registered user with three features: birthdate, gender, and postal code. For the birthdate variable we obtain dates ranging from the year 1895 to 2017. These are converted to years in age for each user, resulting in an age variable ranging from 0 to 123 years old, with a median of 51.

For gender, the user may specify one of three values: 'male', 'female' or 'other'. The acquired totals (38,610, 24,124, and 209, respectively) for each category reveal a larger proportion of male users than female users and a small fraction having chosen 'other'.

In the post code field, users may input any four digits pertaining to their area of residence. The dataset reveals 4,048 different input variations including the case in which the field is left entirely blank. Overall, we have missing values for 547 users, while the top 21 most popular postal codes have approximately 100+ registered users. This is in contrast to the nearly 1000 post codes having just 1 registered user and the remaining post codes which have users varying between 1 and 100.

[2]The data obtained has been anonymized for the purpose of privacy protection.

For the obtained range of values in the variables, we encounter observations for which the information provided is not useful for our problem domain. For the age variable, this means excluding data outside the trusted range of 12 to 100 years old. Furthermore, we consider the gender category Other as having too few observations relative to the Male and Female categories, therefore this category is omitted from further study. In addition, we omit instances where the registered area code is left blank or invalid (e.g. area code registered as '+450'). After filtering, we ultimately remain with 61,933 observations in the demographic dataset.

Having filtered the collected data, the variables are then categorized into their respective groups. For age group we initially[3] consider the age intervals defined by 12-17, 18-24, 25-29, 30-49, 50-66, and 67+. That is, each observation is assigned a corresponding age group for which they belong to. For the geographical variable, each postal code is mapped to what is considered either an 'urban' area or a 'rural' area. Specifically, we define an urban area to refer to one of the four largest cities in Norway with densely populated surrounding areas included – while the term rural refers to all other areas.

The geographical grouping was achieved by using both postal code listings obtained from Posten[4] and information made available by Statistics Norway (SSB) on geography[5]. Particularly, the post codes are grouped by their associated municipalities, e.g. all post codes associating to the Oslo region are grouped as one, while those associated to Bergen are grouped as another. To define 'a densely populated surrounding area' we consider SSB's standards for listing populations[6]. For example, Oslo and surrounding areas consists of the municipalities: Oslo, Ski, Oppegård, Bærum, Asker, Sørum, Rælingen, Lørenskog, Skedsmo, Nittedal, Lier and Røyken.

In Figures 3.3 and 3.5 we display the gender and age distribution in the demographic data. Figure 3.3 shows greater representation across users in the mid-30's to 70's range. For the previously defined 6 age categories this corresponds to a considerable imbalance among the groups. This is illustrated in Figure 3.4 wherein roughly 85% of users in this dataset fall into the 3 oldest age categories, while the remaining are categorized into the 3 younger groups. Similarly, Figure 3.5 displays a significantly greater proportion amongst male users (61.64%) compared to female users (38.36%).

[3]We later work with alternative groupings in age group (cf. Section 4.4 and Section 4.5) to the effect that some observations are assigned an alternative age group instead of the initial assignment.

[4]Posten Norge AS. *Postnummer i Norge*. Accessed March 2018. URL: https://data.norge.no/data/posten-norge/postnummer-i-norge.

[5]Statistisk Sentralbyrå. *Befolkning*. Tabell 11727 via SSB API. Accessed March 2018. 2017. URL: https://www.ssb.no/befolkning/statistikker/folkemengde/aar-per-1-januar#scroll.

[6]Statistisk Sentralbyrå. *Tettsteders befolkning og areal*. Accessed March 2018. 2017. URL: https://www.ssb.no/beftett.

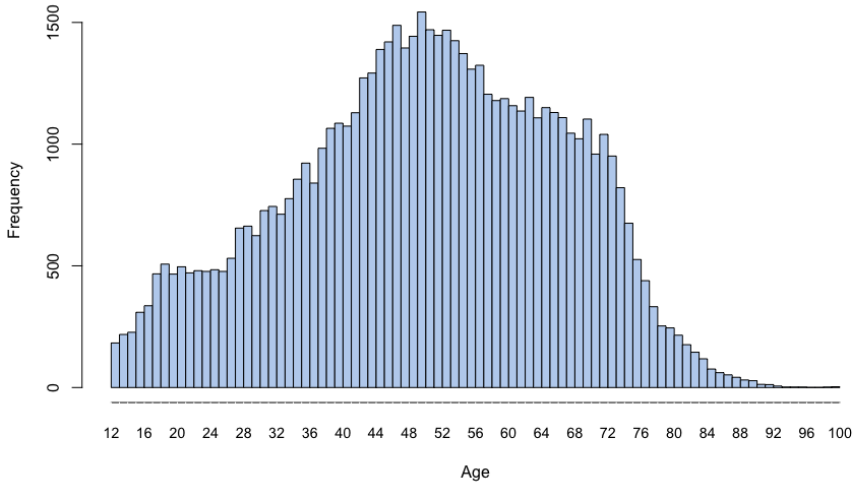


Figure 3.3: Age distribution of logged-in NRK users

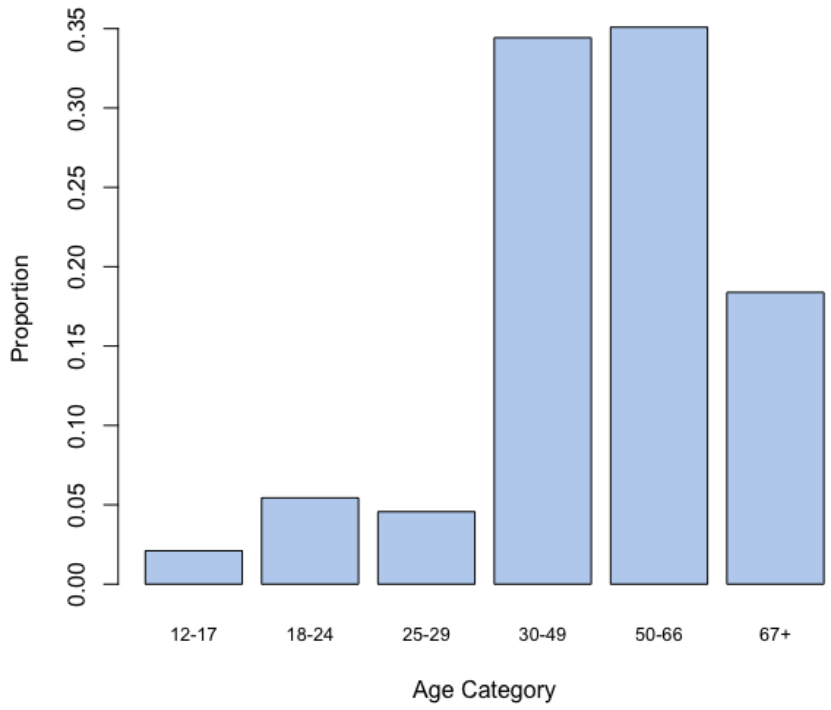


Figure 3.4: Age group distribution of logged-in NRK users, for the 6-class setting.

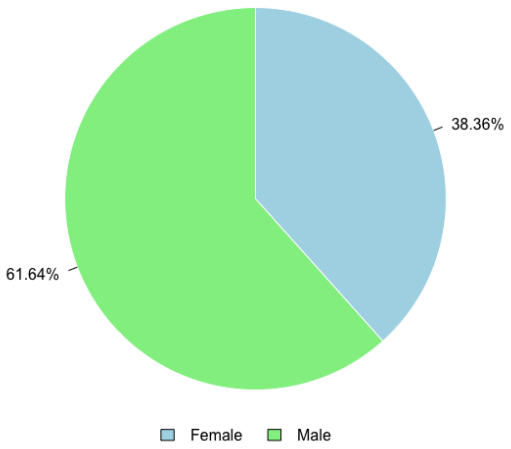


Figure 3.5: Proportions of of female and male logged-in users.

CHAPTER 4

Analysis

In the remainder of this thesis we examine and discuss results of our study. This chapter, in particular, is structured such that the first part (Section 4.1) focuses on assessing NRK's demographic reach within Norway while the second part (Section 4.3 - Section 4.6) is centered around examining the obtained results for predicting user demographics from content consumption behaviour.

The first part addresses Q1 in Section 1.2 by comparing NRK's demographic distribution for logged-in users to Norway's demographics at large. The second part addresses Q2 and Q3 by performing classification experiments using the previously described demographic and content consumption datasets. In doing so, we use demographic groups (age and gender) as response variables and content consumption variables as covariates (cf. Section 3.1.4). Classifier performance is then reported in terms of evaluation metrics previously discussed in Section 2.4. In addition, we group prediction results by the number of unique contents viewed in order to investigate if consuming more unique contents is positively related to prediction accuracy.

In age group classification we initially consider 6 age categories, as previously described. Additionally, however, we choose to further investigate the effects of modifying age categories on prediction quality, since it is not obvious what the best way to categorize age is. We hence try two alternative ways of grouping age. The first consists of 4 classes while the second consists of two. Ultimately we obtain three sets of age prediction results which we use to compare and contrast metrics.

4.1 Assessing NRK's Demographic Reach

In order to assess NRK's demographic reach across Norway, we require relevant population values for Norway as a whole, namely, the number of inhabitants within a given region, for males and females and for each age category. These we obtain from SSB's API service[1], to get an overview of totals such as the one displayed in Table 4.1.

In obtaining known population values for Norway, we may assess, based on NRK's demographic data on digital users, NRK's reach for logged-in users

[1] https://www.ssb.no/en/omssb/tjenester-og-verktoy/api

	Female	Male
12-17	184,059	193,825
18-24	228,852	245,404
25-29	182,022	189,553
30-49	702,065	741,387
50-66	536,721	554,249
67+	428,040	358,319

Table 4.1: Norway's population totals for given age groups

	Female	Male
12-17	734	478
18-24	1703	1646
25-29	1399	1412
30-49	8235	12965
50-66	7929	13891
67+	3747	7800

Table 4.2: NRK's sample totals for given age groups

across age group, sex, and geography. To achieve this we compare the sample data at hand to the known values for the Norwegian population, determining the degree to which the computed sample proportions and the population proportions exhibit similar characteristics.

First, we examine the age group proportions for each gender separately for both NRK and Norway. This is visualized in Figure 4.1 in the form of pie graphs. The top panel displays the proportions of female and male NRK users by age group, while the bottom panel correspondingly shows the known population proportions for Norway as a whole.

A brief examination between the two sets of data shows that the sample audience, to some extent, reflects the Norwegian population distribution across the six different age categories for both males and females. This is particularly evident in that the proportion of NRK users is greatest amongst the older age groups, while less so for the younger groups, which is generally consistent with the Norwegian population distribution in the lower panel. This suggests a certain congruity between the sample data presented and the population at large.

Closer comparison of the proportions between NRK users and the Norwegian population reveals that the age group 30-49 in both the female and male cases are relatively well represented. That is to say that the compared proportions lie close to one another, deviating by less than 5% in both cases. For instance, for female users 34.75% are within this age category, comparable to the 31.04% for the Norwegian female population. The values for the male case are similar, with a ratio of 34.08 to 32.48. Examining the differences in proportions for the

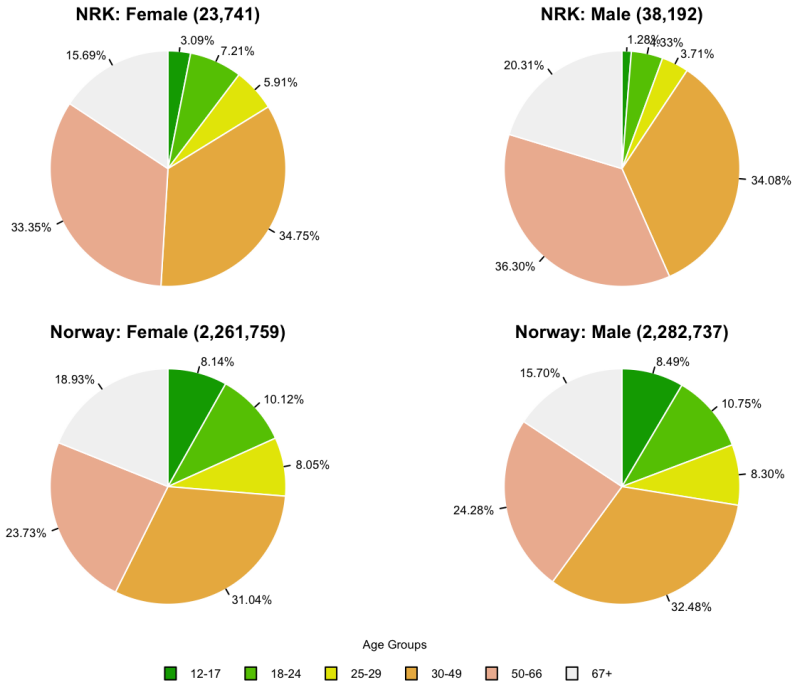


Figure 4.1: Proportions by age group for NRK users (top) and Norway (bottom).

age group 67+, we see that more or less the same applies, however the female counterpart is slightly underrepresented in the NRK sample, while the converse is true for males. For the age group 50-66 we see that the NRK user proportion is well above that of the Norwegian population proportion for both males and females (33.35% to 23.73% for females and 36.30% to 24.28% for males).

Moreover, we see rather prominently that the proportion of male NRK users in the youngest age group 12-17 is well below that of Norway's population proportion (1.28% to 8.49%). Generally, we see a tendency for the younger age groups to be underrepresented in roughly the same manner. This is the case for the male age groups 18-24 and 25-29. It is also further exemplified in the female case where 3.09% of the NRK users are in the age group 12-17, while the proportion for Norway is more than twice that at 8.14%. The differences are less apparent for females in the age categories 18-24 and 25-29, however the underrepresentation is still present.

Subgrouping the data geographically into urban and rural users in Figures 4.2 and 4.3, we see a general reflection of the previous findings, whereby the older age groups have a tendency to be better represented than the younger. Here however we also see, by comparing the relative proportion sizes, a subtle difference revealed in that the urban female in age category 67+ is perhaps better represented than the rural female in this age group. That is, the proportion

of urban females 67+ constituting NRK users (16.47%) is relatively close to the corresponding Norwegian population proportion for that demographic (15.89%) in comparison to that of the rural counterpart (15.13% to 20.54%).

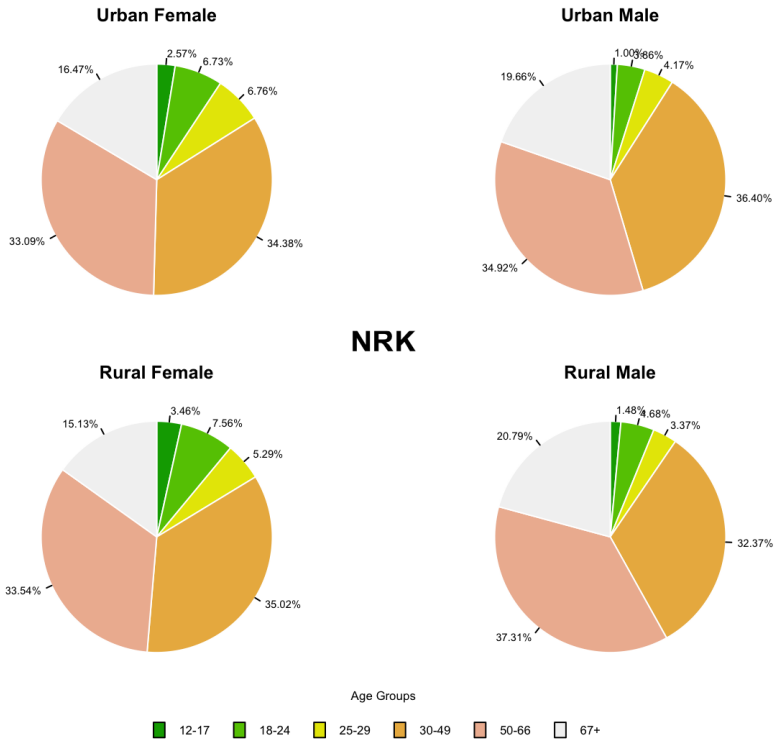


Figure 4.2: Proportions by age category for NRK users, subdivided into an urban and rural group.

Overall, although we see a certain congruity between the NRK sample distribution and the Norwegian population as a whole – and in this way a certain representativeness – we also see an important deviation characterized by an underrepresentation among younger age groups. This in turn seems to be compensated by the older age groups. Moreover, the data suggests that room for improvement in reach for logged-in users is greatest amongst younger males (urban and rural), young females in the age group 12-17, as well as rural females in the age group 67+. These findings can also be summarized by the bar graphs shown in Figure 4.4.

Though the differences in distribution between the NRK sample and Norway's population are perhaps clearly distinguishable on the pie graphs, we may also quantify the notion of representativeness or similarity using a goodness of fit test. To this end, we use the chi-square goodness of fit test where the hypotheses are as follows,

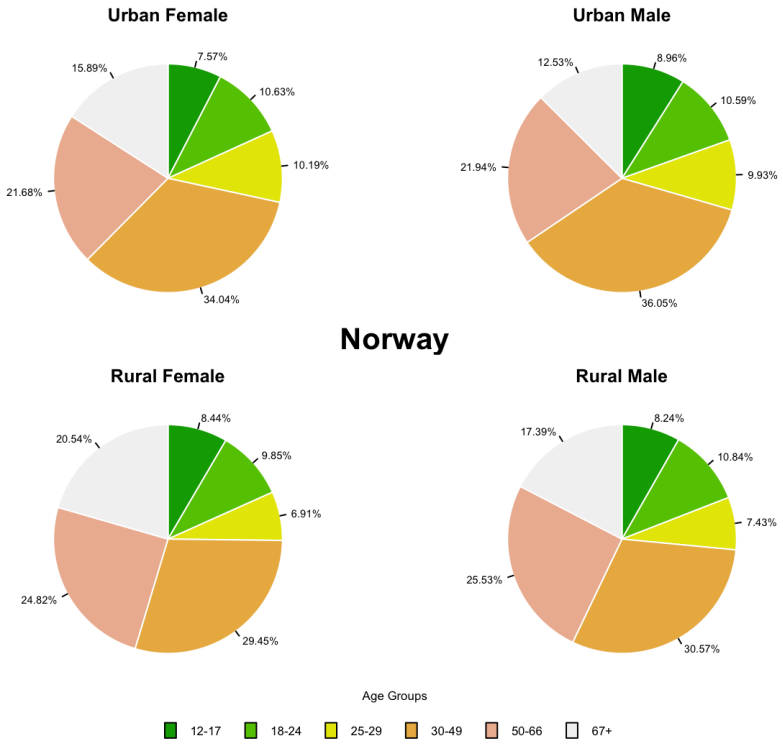


Figure 4.3: Proportions by age category for Norway, subdivided into urban and rural regions.

H_0 : the sample data (NRK's demographic data) fits the specified distribution (here Norway's population distribution).

H_a : the sample data does not fit the specified distribution.

Performing the test procedure for the joint distribution of gender and age group, we obtain, `p-value < 2.2e-16`. This indicates strong evidence for rejecting the null hypothesis in favour of the alternative hypothesis. This verifies that there is indeed a statistically significant difference between the two distributions, as suspected. Repeating the same procedure for the case in which the urban and rural population distributions are also considered yields similar results.

4.2 Preparing Training and Test Sets

Before presenting the results of predicting demographics from content consumption behaviour, we describe in this section the use of training and test sets, and in particular how they were constructed for the classification models.

First, we describe the technicalities involved in the process of building the models. The classifiers used in both age group prediction and gender prediction

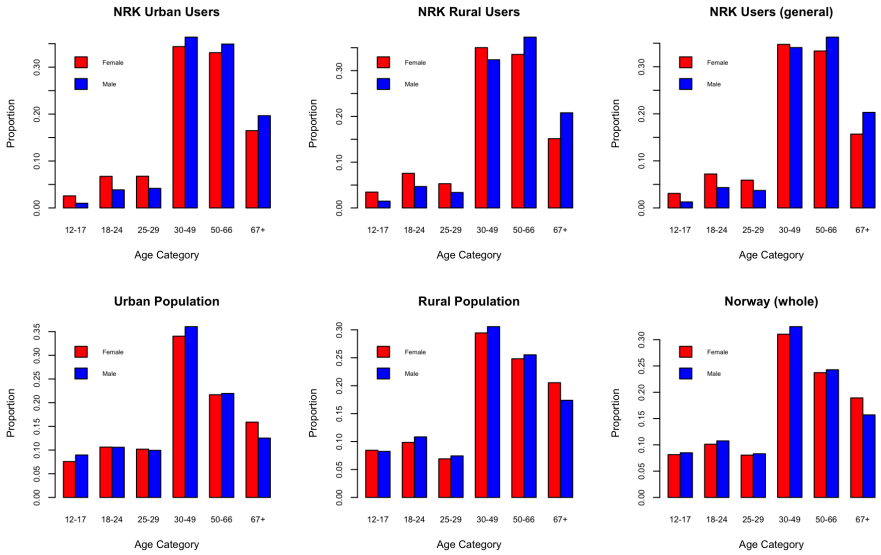


Figure 4.4: (Top) Proportions of NRK users: urban, rural and general. (Bottom) Norway's population proportions: urban, rural and as a whole.

are regression methods (logistic for binary response, multinomial for multiclass response), regularizations of the relevant regression method (both ridge and lasso), KNN, and random forest. Each of these classifiers are trained and evaluated using the holdout method with a total of $K = 10$ runs. Final conclusions for each classifier's performance are then drawn based on an average of the 10 runs.

For each of the $k = 1, ..., 10$ runs the following is performed. Model-fitting begins by randomly splitting the dataset into a training sample and test sample. The training set, in particular is selected such that the output variable (age group or gender) is completely balanced. This is to avoid having the minority classes overpowered by the majority class – which in our case is of special concern due to the significant class imbalance (Section 3.2). The test set, in contrast, is balanced according to the proportions of the original dataset with the assumption that the sample we have obtained is representative of NRK's user base.

To obtain a balanced training sample for each of the $k = 1, ..., 10$ runs, we first identify the smallest class size, n_i, in the entire working dataset. The sizes of each class in the training sample is then chosen to be 80% of n_i, the identified smallest class. This then, through random sampling, generates the training set for a given run k. The corresponding test set is generated using the remainder of the data that does not appear in the training set. This is done by identifying the class sizes in the remaining data and adjusting them such that the class proportions equate to that of the original dataset. For the adjusted test classes this is also achieved through random selection. Together these define the training and test sets for a given run k.

For predicting age group, which in our case initially consists of 6 classes: 12-17, 18-24, 25-29, 30-49, 50-66, and 67+, the full dataset has class sizes and proportions that are displayed in Table 4.3.

	12-17	**18-24**	**25-29**	**30-49**	**50-66**	**67+**
Class size	749	1886	1604	10304	8733	4295
Class proportion	0.03	0.07	0.06	0.37	0.32	0.16

Table 4.3: Initial class distribution for predicting age group in our 6-class setting.

The process of selecting training and test sets for each run results in class sizes as displayed in Table 4.4. Each training sample is a balanced set with $749 \cdot 0.8 \approx 599$ observations for each age category and a total of 3,594 observations. The test sets consist of 5,520 sampled observations independent of training set and with age categories that follow the proportions of the original dataset.

	12-17	**18-24**	**25-29**	**30-49**	**50-66**	**67+**
Training set	599	599	599	599	599	599
Test set	150	377	321	2,063	1,749	860

Table 4.4: Training and test set class sizes for the 6 class age-group prediction problem.

For gender prediction, the minority group is females with 10,487 observations in the entire working dataset. Table 4.5 displays the class sizes and proportions in the gender classification setting. Following the same procedures for selecting training and test sets, we arrive at class sizes as shown in Table 4.6. Balancing the class sizes results in training sets with 8,390 observations in both groups to constitute a total of 16,780. Corresponding test sets then consist of 5,513 observations in total, with 2,097 in the female group and 3,416 in the male group.

	Female	**Male**
Class size	10,487	17,084
Class proportion	0.38	0.62

Table 4.5: Initial class distribution in gender prediction.

	Female	**Male**
Training	8,390	8,390
Test	2,097	3,416

Table 4.6: Training and test set class sizes for gender classification.

After training and test sets are selected, models are fitted to the data in the training sets. For the methods involving tuning parameters, 10-fold cross-

validation is implemented in each training run to obtain tailored parameter values for each of the 10 holdout sets.

4.3 Classification with Six Age Groups

We now go on to present results for the prediction study. We begin here with predicting logged-in users into one age category out of six (as previously defined: 12-17, 18-24, 25-29, 30-49, 50-66, 67+), before proceeding with a 4-class and binary class setting in the sections that follow.

Tables 4.7 - 4.12 provide details on overall classifier performance in the 6-class prediction case. Table 4.7 shows the overall baseline accuracy along with the overall accuracies achieved by the individual methods. Tables 4.8 - 4.12 each report the resulting recall, precision, and F_1-score values by age group for each classifier.

From Table 4.7, we see that all five methods perform worse than the baseline reference method in terms of accuracy. Among the five classifiers the highest accuracy obtained is by multinomial regression at 33.7%. A confusion matrix corresponding to this classifier can be found in Appendix A. The regularization methods follow after, performing identically with 33.3% for ridge and 33.4% for lasso. Random forest performs slightly worse with an accuracy of 33.0%, while KNN has the lowest value at 24.2%. For random forest, lasso, ridge, and multinomial regression the accuracies across the 10 runs are similarly dispersed with standard deviations of approximately 0.01. KNN obtains slightly less spread values with a standard deviation of 0.004.

In terms of precision, recall, and F_1-score, we see similar patterns across the five classifiers. More specifically, recall is generally highest for age groups 12-17 and 67+, while lower for the remaining four age groups. This indicates that the classifiers are able to identify a larger proportion of the youngest and oldest age groups compared to the remaining four. We obtain the highest recall for age group 12-17, with values ranging from 0.487 (KNN) to 0.612 (lasso). Age group 30-49 has the lowest recall for the regression methods and KNN, and second lowest for random forest. The recall values for this age group range from 0.192 (KNN) to 0.259 (multinomial regression).

For precision, age group 30-49 is consistently highest across all methods with values ranging from 0.432 (KNN) to 0.537 (lasso). This seems consistent with the trade-off nature between recall and precision [6]. Since the classifiers identify a lower amount of 30-49 users, it is relatively easier to obtain a higher proportion of relevant predictions within this class. Furthermore, this tells us that, in terms of exactness, approximately 50% of classifications into this class are valid.

The lowest precision we obtain is for age group 12-17 with a range of 0.070 (KNN) - 0.125 (multinomial regression). We also see that the groups 18-24 and 25-29 have similarly low precision, while 67+ and 50-66 have more moderate values. For 50-66 we obtain a range of 0.412 (KNN) to 0.485 (multinomial

regression) and for 67+ we obtain a range of 0.310 (KNN) to 0.360 (multinomial regression). This tells us that, for the older age groups we tend to obtain more valid predictions, and hence better exactness, compared to the younger.

Examining F_1-score we see a similar occurrence where the quality of predictions seem to be better for the older age groups than for the younger. In particular, the 25-29 group consistently has the lowest range of score (0.119 (KNN) - 0.173 (lasso and multinomial regression)), and conversely 67+ consistently has the highest range (0.355 (KNN) - 0.434 (multinomial regression)). This means that when recall and precision are combined – with equal importance – into one measure, the classification performance for the group 67+ surpasses that of 25-29.

Classifier	Accuracy (SD)
Multinomial regression	0.337 (0.010)
Ridge regression	0.333 (0.011)
Lasso regression	0.334 (0.010)
KNN	0.242 (0.004)
Random forest	0.330 (0.007)
Baseline	0.374

Table 4.7: Overall accuracies for predicting age group with 6 classes.

Multinomial Regression

Age group	Recall (SD)	Precision (SD)	F_1 (SD)
12-17	0.600 (0.036)	0.125 (0.008)	0.206 (0.012)
18-24	0.271 (0.013)	0.172 (0.009)	0.211 (0.008)
25-29	0.277 (0.026)	0.126 (0.013)	0.173 (0.017)
30-49	0.259 (0.025)	0.532 (0.015)	0.347 (0.024)
50-66	0.328 (0.024)	0.485 (0.013)	0.391 (0.018)
67+	0.547 (0.029)	0.360 (0.011)	0.434 (0.012)

Table 4.8: Recall, precision, and F_1-score for predicting age group with 6 classes using multinomial regression.

The results of grouping prediction performance by the number of unique contents viewed by users is displayed in Table 4.13 and Figure 4.5. The quantity of unique contents is defined in 10 intervals (e.g. users who have viewed 3-8 unique contents belong in one group, those who have viewed 9-15 belong in another, etc.) and chosen such that all intervals contain the same amount of users. This is achieved by identifying the cut points in the range of unique contents viewed that divide the users into 10 equal groups. In Table 4.13 and Figure 4.5 we report the intervals of quantity viewed along with the obtained overall accuracy pertaining to each group. To get the reported values, we apply the best performing classifier in terms of accuracy – which in this case is multinomial regression. We also note that in the sections that follow (Sections 4.4-4.6), the same procedures are applied to obtain analogous results for group-

Ridge Regression

Age group	Recall (SD)	Precision (SD)	F_1 (SD)
12-17	0.609 (0.043)	0.120 (0.008)	0.201 (0.013)
18-24	0.265 (0.016)	0.173 (0.012)	0.209 (0.012)
25-29	0.270 (0.022)	0.126 (0.011)	0.171 (0.015)
30-49	0.252 (0.027)	0.531 (0.016)	0.341 (0.027)
50-66	0.329 (0.024)	0.484 (0.014)	0.391 (0.017)
67+	0.543 (0.029)	0.353 (0.009)	0.428 (0.010)

Table 4.9: Recall, precision, and F_1-score for predicting age group with 6 classes using ridge regression.

Lasso Regression

Age group	Recall (SD)	Precision (SD)	F_1 (SD)
12-17	0.612 (0.040)	0.122 (0.008)	0.204 (0.012)
18-24	0.259 (0.014)	0.170 (0.012)	0.205 (0.012)
25-29	0.273 (0.020)	0.127 (0.012)	0.173 (0.015)
30-49	0.253 (0.024)	0.537 (0.014)	0.344 (0.023)
50-66	0.327 (0.024)	0.480 (0.011)	0.389 (0.017)
67+	0.548 (0.025)	0.353 (0.009)	0.429 (0.009)

Table 4.10: Recall, precision, and F_1-score for predicting age group with 6 classes using lasso regression.

KNN

Age group	Recall (SD)	Precision (SD)	F_1 (SD)
12-17	0.487 (0.045)	0.070 (0.006)	0.123 (0.010)
18-24	0.223 (0.020)	0.112 (0.008)	0.149 (0.010)
25-29	0.206 (0.027)	0.084 (0.008)	0.119 (0.012)
30-49	0.192 (0.013)	0.432 (0.014)	0.266 (0.014)
50-66	0.206 (0.017)	0.412 (0.019)	0.274 (0.015)
67+	0.415 (0.013)	0.310 (0.010)	0.355 (0.008)

Table 4.11: Recall, precision, and F_1-score for predicting age group with 6 classes using KNN.

ing predictions in terms of unique contents.

Overall we see a trend of increase in accuracy as the quantity of unique contents increases. Here, we obtain the lowest accuracy when the user has only seen between 3 to 8 unique contents and in contrast, the highest accuracy for the interval 617+. Moreover, we see that in order to achieve accuracy greater than the baseline accuracy of 0.375, users generally have to have viewed 171 or more unique contents, which seems to be quite a lot of content. The multinomial regression classifier also seems to perform better for the subset of users who have viewed 104+ compared to the multinomial regression result obtained in Table 4.7.

Random Forest			
Age group	Recall (SD)	Precision (SD)	F_1 (SD)
12-17	0.565 (0.037)	0.128 (0.008)	0.208 (0.013)
18-24	0.248 (0.010)	0.164 (0.010)	0.197 (0.009)
25-29	0.274 (0.031)	0.118 (0.006)	0.165 (0.011)
30-49	0.253 (0.022)	0.517 (0.021)	0.339 (0.022)
50-66	0.336 (0.017)	0.472 (0.010)	0.392 (0.012)
67+	0.519 (0.026)	0.347 (0.009)	0.416 (0.012)

Table 4.12: Recall, precision, and F_1-score for predicting age group with 6 classes using random forest.

Nr. of unique contents	**Accuracy**
3-8	0.225
9-15	0.318
16-25	0.280
26-41	0.297
42-64	0.305
65-103	0.304
104-170	0.361
171-311	0.383
312-616	0.435
617+	0.469

Table 4.13: Binning results for age classification with 6 groups

4.4 Classification with Four Age Groups

We investigate the effects of an alternative age grouping, as it is difficult to know how good the initially defined categorization is. To this end, we consider the results of merging the three youngest age groups to obtain four classification labels, namely, 12-29, 30-49, 50-66, and 67+. This gives us the advantage of having more data in the merged group compared to having three individual groups.

Table 4.14 displays the overall accuracy obtained by each method. We see from this table that the regression classifiers along with random forest perform identically, having obtained accuracies of 0.431, with standard deviations of 0.005. All three outperform the baseline method by 15.2% which has an accuracy of 0.374. This is contrast to KNN which underperforms (0.363) relative to the baseline.

The multinomial regression results for recall, precision, and F_1-score are displayed in Table 4.15. A confusion matrix corresponding to this classifier can be found in Appendix A.2. The remaining tables for the other regression methods and random forest, which perform similarly, can be found in Appendix A.2, along with KNN, which generally performs worse.

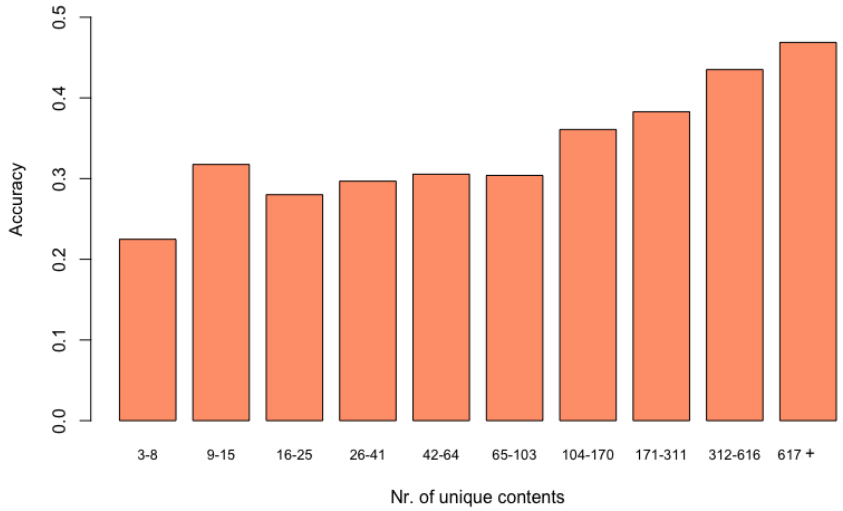


Figure 4.5: Binning results for age classification with 6 groups

From Table 4.15, we see that age group 12-29 and 67+ yield the highest recall values (0.701 and 0.568 respectively). Particularly for the class 67+, we see an improvement from 0.547 in Table 4.8, when using 6 age groups, to 0.568 in recall. We also see a slight increase for age group 50-66 from 0.328 to 0.332. An even larger improvement takes place for age group 30-49, which previously had a recall of 0.259, and increases here to 0.345. These improvements suggest that the classifiers capture more instances of these classes when using 4 age groups compared to 6 age groups.

For precision, age groups 30-49 and 50-66 yield higher values (0.535 and 0.481 respectively) than the youngest and oldest age groups, 12-29 and 67+ (0.369 and 0.357 respectively). This indicates that we are able to obtain more exact predictions for the two middle age groups compared to the youngest and oldest groups. For the two oldest age groups (50-66 and 67+), precision values have gone down slightly from Table 4.8 to Table 4.15 (0.485 and 0.360 to 0.481 and 0.357 respectively).

Considering both recall and precision combined, the lowest achieved F_1-score belongs to the group 50-66 (0.393), while the highest is achieved by 12-29 (0.483). We also see a general improvement in values for the 3 oldest classes, compared to Table 4.8. The improvement is not as substantial for 50-66 and 67+, however the age group 30-49 has improved from 0.347 to 0.419. The improvement from the 6-class case suggests the strongest improvement of prediction quality is for age-group 30-49. We also observe that the quality of predictions across age groups are not too spread apart, all four being in the 0.4 range. Furthermore, it is worth noting that standard deviations have also generally decreased compared

to Section 4.3, indicating less dispersion and more stable results.

Classifier	Accuracy (SD)
Multinomial regression	0.431 (0.005)
Ridge regression	0.431 (0.005)
Lasso regression	0.431 (0.005)
KNN	0.363 (0.006)
Random forest	0.431 (0.005)
Baseline	0.374

Table 4.14: Overall accuracies for predicting age group with 4 classes.

Multinomial regression

Age group	Recall (SD)	Precision (SD)	F_1 (SD)
12-29	0.701 (0.017)	0.369 (0.010)	0.483 (0.012)
30-49	0.345 (0.010)	0.535 (0.013)	0.419 (0.009)
50-66	0.332 (0.015)	0.481 (0.009)	0.393 (0.012)
67+	0.568 (0.013)	0.357 (0.008)	0.439 (0.009)

Table 4.15: Recall, precision, and F_1-score for predicting age group with 4 classes using multinomial regression.

The results of grouping predictions by quantity of unique contents viewed for the 4-class problem is displayed in Table 4.16 and Figure 4.6. The method used to obtain the results was multinomial regression. As in Section 4.3, we see a relatively steady increase in accuracy with the number of unique contents viewed by users. The lowest accuracy (0.299) is observed with the lowest interval of unique contents (3-8), while the highest (0.567) is achieved at the interval 617+. Table 4.16 shows that in the case of 4 age groups, the baseline accuracy is surpassed after approximately 26-41 unique contents viewed. This is a substantial improvement from the 6-class results which required 171-311 unique contents. Furthermore we observe that, in order to achieve better classification results than the overall results in Table 4.14, unique contents viewed should be in the interval 104-170 or higher.

4.5 Binary Age Group Classification

In addition to predicting with 6 and 4 age groups, it is also interesting for NRK to see how well prediction works in a binary age group setting, namely with the class labels 12-39 and 40+. In this section, we thus present the results of classification given these two age groups as the response categories.

Table 4.17 displays the obtained overall accuracy values for the different classifiers. Similar to the examined results in Section 4.4 and Section 4.3, KNN also performs worst out of all five methods with overall accuracy at 63.7%. Unlike previously seen, however, random forest performs best with overall accuracy of 71.9%. A confusion matrix corresponding to this classifier can be found in

Nr. of unique contents	Accuracy
3-8	0.328
9-15	0.413
16-25	0.361
26-41	0.395
42-64	0.427
65-103	0.419
104-170	0.446
171-311	0.472
312-616	0.494
617+	0.563

Table 4.16: Binning results for age classification with 4 groups.

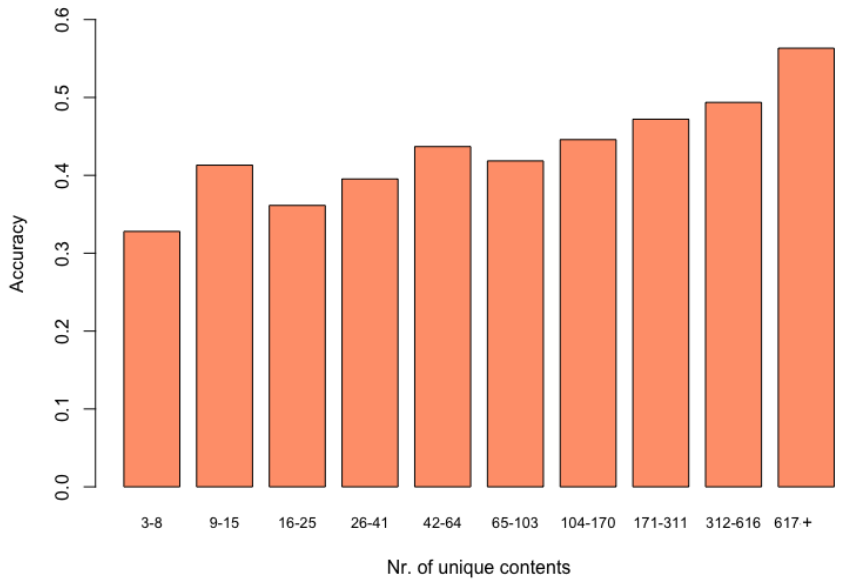


Figure 4.6: Binning results for age classification with 4 groups.

Appendix A.3. Though all three regression models perform only slightly worse with identical values of 71.2%. We also see that using regression and random forest we can obtain slightly more accurate results than simply classifying to the most frequently occurring class (baseline accuracy: 69.7%). Particularly for random forest we see a 3.2% improvement from the baseline accuracy. KNN, on the contrary, does not outperform the baseline method.

Having obtained the highest overall accuracy and knowing that the other models generally performed worse, we present the recall, precision and F_1-score for random forest in Table 4.18, while the remaining tables are found

in Appendix A.3. For age groups 12-39 and 40+ we obtain recall values of 0.737 and 0.712 respectively, indicating a better ability to capture instances of 12-39 users than 40+. In terms of precision, age group 40+ achieves a higher proportion than 12-39 (0.862 and 0.526 respectively). This suggests more exact classifications for the former class compared to the latter.

The F_1-score shows a higher value of 0.780 for predictions in the 40+ age group than the 12-39 age group, which obtained a value of 0.614. This means that when recall and precision are combined and given equal importance, classification performance is generally favourable for the 40+ age group compared to the 12-39 age group. In addition, the standard deviation values displayed in both Tables 4.17 and 4.18 suggest low dispersion among the 10 runs, as in Section 4.4.

Classifier	Accuracy (SD)
Logistic regression	0.712 (0.005)
Ridge regression	0.712 (0.006)
Lasso regression	0.712 (0.005)
KNN	0.637 (0.007)
Random forest	0.719 (0.005)
Baseline	0.697

Table 4.17: Overall accuracies for binary age group prediction.

Random Forest			
Age group	Recall (SD)	Precision (SD)	F_1 (SD)
12-39	0.737 (0.009)	0.526 (0.007)	0.614 (0.005)
40+	0.712 (0.008)	0.862 (0.004)	0.780 (0.005)

Table 4.18: Recall, precision, and F_1-score for binary age group prediction using random forest.

Table 4.19 shows prediction accuracy by intervals of unique contents viewed, achieved by random forest. The lowest accuracy is 0.659 when the number of unique contents is in the interval 3-8. The highest accuracy obtained is 0.771 for the interval 104-170. An overall positive relationship is not as prominent in this case as in the 6- and 4- class problems. As Figure 4.7 illustrates, accuracy seems to fluctuate throughout the quantity intervals. To some degree, accuracy increases up to the 104-170 interval, after which it begins to decline. We also observe that most intervals achieve an accuracy greater than that of the baseline of 0.697, except for a few (3-8, 26-41, and 617+).

4.6 Gender Classification

In this section, we present the results of classifying test users by gender. Like in previous sections, we display overall accuracy for all methods, along with recall, precision, and F_1-score for the best performing classifier (random forest).

Nr. of unique contents	Accuracy
3-8	0.659
9-15	0.713
16-25	0.700
26-41	0.690
42-64	0.723
65-103	0.731
104-170	0.771
171-311	0.755
312-616	0.721
617+	0.685

Table 4.19: Binary age classification accuracy by quantity of unique contents viewed.

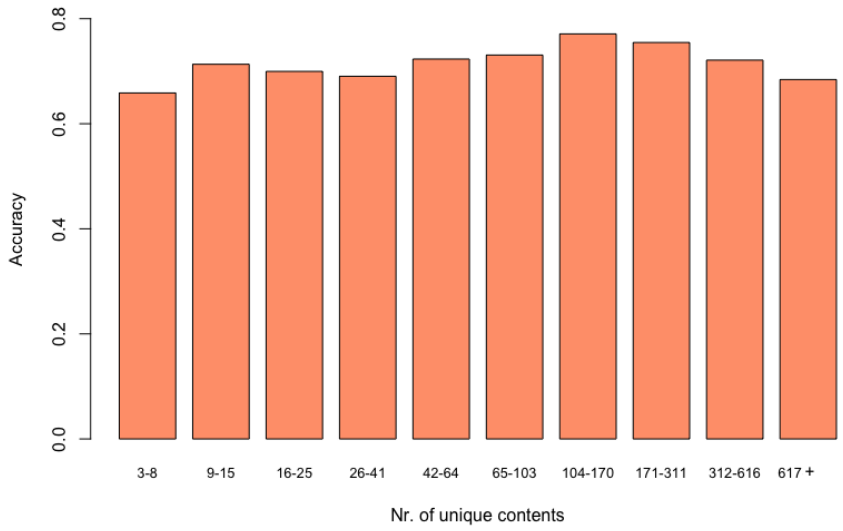


Figure 4.7: Binary age classification with predictions grouped by quantity of unique contents viewed.

These can be seen in Tables 4.20 and 4.21. Logistic regression, ridge, lasso, and KNN follow similar patterns in terms of recall, precision and F_1-score and can therefore be found in Appendix A.4. Also found in Appendix A.4 is a confusion matrix corresponding to the random forest classifier.

Examining accuracy in Table 4.20, we see that not unlike the previous classifications, KNN performs the poorest at 62.8% accuracy. Nonetheless, it is able to achieve higher accuracy than the baseline, which is at 62.0%. The regression classifiers and random forest all perform similarly, achieving approximately 73% accuracy. Similar to Section 4.5, we obtain the highest overall accuracy for

random forest (72.7%), followed closely by logistic regression (72.6%). Further, we see that random forest is able to outperform the baseline by 17.3%.

In terms of recall, the random forest classifier detects male and female users with similar accuracy, though it detects females with a slightly higher accuracy. This is shown in Table 4.21 where under recall the male category has a value of 0.725, while the female counterpart has 0.730.

Conversely, random forest predicts with greater validity for males than females, as reflected by the male group achieving a higher precision (0.814 vs. 0.619). The overall quality of predictions, based on both recall and precision is better for the male category than the female category. This is reflected in the F_1-scores which is 0.767 for males and 0.670 for females.

Classifier	Accuracy (SD)
Logistic regression	0.726 (0.004)
Ridge regression	0.723 (0.003)
Lasso regression	0.725 (0.004)
KNN	0.628 (0.004)
Random forest	0.727 (0.003)
Baseline	0.620

Table 4.20: Overall accuracies for gender prediction.

Random Forest			
Gender	Recall (SD)	Precision (SD)	F_1 (SD)
Male	0.725 (0.005)	0.814 (0.003)	0.767 (0.003)
Female	0.730 (0.006)	0.619 (0.004)	0.670 (0.004)

Table 4.21: Recall, precision, and F_1-score for gender prediction using random forest.

Table 4.22 and Figure 4.8 display, for gender classification with random forest, the results of grouping predictions based on number of unique contents viewed by users. Examining these results, we see a tendency for accuracy to increase with unique contents viewed. The lowest accuracy obtained is 0.644, which corresponds to the lowest interval of unique contents viewed, 3-8. Conversely, the highest accuracy (0.809) belongs to the interval with largest quantity of unique contents (617+). We observe that even with only 3-8 unique contents viewed, we are able to outperform the baseline accuracy of 0.620. We also see that after 42-64 unique contents, the accuracy generally surpasses the achieved values in Table 4.20.

Nr. of unique contents	Accuracy
3-8	0.644
9-15	0.657
16-25	0.728
26-41	0.749
42-64	0.727
65-103	0.751
104-170	0.760
171-311	0.771
312-616	0.753
617+	0.809

Table 4.22: Gender classification accuracy by quantity of unique contents viewed.

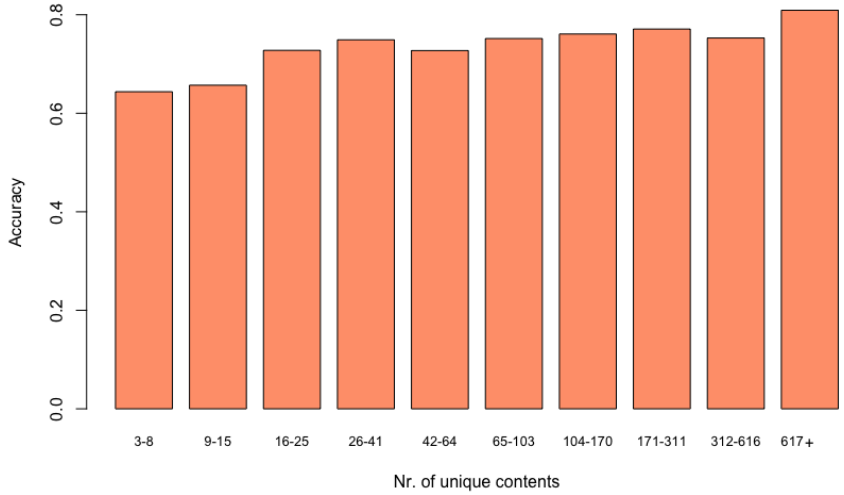


Figure 4.8: Gender classification with predictions grouped by quantity of unique contents viewed.

CHAPTER 5

Discussion and Conclusion

The main objective of this thesis has been to analyze the demographics of NRK's logged-in user base. In light of this, we sought out to answer three research questions (cf. Section 1.2). The first question, Q1, examines NRK's demographic reach among the Norwegian population. The remaining two questions, Q2 and Q3, explore the degree to which prediction methodology can be applied to the NRK data.

Q1 was addressed by comparing differences and similarities in NRK's sample demographics to that of Norway's as a whole. This allowed us to evaluate whether or not the logged-in users are representative of the Norwegian population, with respect to age group, gender, and rural/urban living. Evaluating representativeness then helped us to determine where there is potential room for improvement in users. Our findings suggest two main key points. The first is that, there are clearly underrepresented demographic groups as evidenced by the larger disparities between the Norwegian population proportions and NRK's sample proportions. The second point is that, despite the obvious differences between the two sets, there also exist congruencies in the general form of the data. This was, for example, seen in how the older age groups constituted the largest fraction for both NRK and Norway as a whole, while the younger formed only small fractions.

Moreover, our findings about which demographic groups are underrepresented and which are well represented give NRK insight as to where reach might be improved. Through comparison we established that, in general, younger age groups, both male and female, are potential groups for improvement in terms of reach.

A limitation on the assertions we make about NRK's reach lies in our inability to verify the available data. In particular, there is no way to verify whether or not the logged-in user dataset is truly representative of NRK's user base as a whole. In this regard, our assertions are relative to those who have chosen to provide information, and groups who may be more skeptical to doing so are likely not represented. This may be a reflection of the small portion of young users compared to the older.

To answer Q2 and Q3, we used learning methods to model demographics as functions of content consumption variables. The models were fitted to training

data and then used to predict the demographics of test users, namely age group and gender. Predicting gender was somewhat successful as the classifiers were able to outperform the baseline in terms accuracy and quality of predictions. Random forests in particular was able to achieve an overall accuracy 17.3% higher than the baseline. Predicting age group for the 6-class setting proved to be more challenging, where the highest overall accuracy achieved was approximately 34%, compared to a baseline accuracy of 37%. Predicting with 4-age groups, however, proved to show an improvement, yielding an overall accuracy that was 15.2% better than the baseline. In general however, classification with 4 and 2 classes, were more successful as both were able to obtain accuracies above the baseline for most methods. Furthermore, results by age group varied considerably, with some obtaining lower F_1-scores than others. This was particularly evident in the 6 age group case and slightly less so for predicting gender, and age group with 4 and 2 classes.

Among other findings, we saw that KNN consistently performed the worst compared to the other classifiers. This can possibly be attributed to the number of features (36) in our design matrix, as KNN tends to suffer in higher dimensions [18]. We also saw that both ridge and lasso achieved nearly identical, if not lower, values than multinomial regression. This suggests that overfitting was not necessarily a problem, and hence the effects of regularization may be negligible. In addition we found that random forest and multinomial regression had the tendency to perform the best.

Q3, specifically, was answered by grouping predictions based on the number of unique contents viewed by the logged-in user and determining the fraction of correctly classified users within those groups. In three out of the four classifications that were performed, we saw a tendency for the accuracy to improve with the quantity of unique contents viewed. The exception case occurred in the binary age classification, where instead of a positive relationship, we saw fluctuations in accuracy. For the case of 6-age groups, the baseline accuracy was exceeded at the interval of 171-311 unique contents, while for the 4-age groups case this was 26-41. In gender classification, the lowest interval, 3-8, was able to produce a higher accuracy than the baseline.

5.1 Prediction Challenges

Our classification study reveals that in practice, obtaining accurate prediction results can be rather challenging, most especially for a multi-class problem. The accuracies achieved by the models reflects the necessity to improve on certain modelling aspects.

Recalling from Chapter 2 that our methodological approach involved balancing our training data through random undersampling, it is a possibility that our methods suffered the loss of important data points. Since implementing random undersampling involves discarding random training observations to obtain all classes of equivalent size, this meant for us discarding a substantial amount of available data due to the significant imbalance. Despite having several thousand rows of observations to begin with, we ultimately use only a

fraction of these (cf. Section 4.2 and Chapter 3). Applying other, more clever techniques to avoid losing useful information in the process of balancing could have resulted in having a greater number of training instances and perhaps improved classification accuracy.

The ability to predict the demographics of users may be particularly dependent on how the response variable is categorized. This is especially relevant for the age group classification problem and less so for gender classification, as the defining characteristics of each category may not be as clear-cut. As an example, the low accuracy obtained from classifying with 6 age groups could stem from strong similarities between the younger age groups. If, for instance, users in the 18-24 category exhibit similar viewing behaviour as those in 25-29, the learning algorithms may have the tendency to confuse whether a test observation belongs in one or the other. In order to obtain better predictive ability, an option may be to choose more meaningful age intervals that are easier to distinguish between. This is not as straight forward as it may seem, as it raises the question of how one might go about choosing a cutoff.

The inherent nature of the content consumption data may also be a direct limitation on prediction quality. The Collaborative Filtering variables, for example, are based on implicit feedback – which is, by nature, noisy [13]. This means that the Collaborative Filtering run that produced our data could have a particularly low signal-to-noise ratio. Ultimately this makes modelling the complexity of viewing preference more challenging in practice than theory, as possibly reflected in our inability to substantially outperform the baseline method.

A specific source of uncertainty is the premise that the information extracted about user behaviour actually reflects the individual who maintains the NRK account, as opposed to several people. There is a certain level of ambiguity in terms of who is actually watching an episode or program, which in itself is difficult to assess. In light of this fact, we took the measure of removing TV device entries (cf. Chapter 3). This however does not completely eliminate the possibility that individuals may still share other devices. This is an issue that is difficult to verify. One option to address it may be to repeat a similar study limited to mobile devices, as these are largely more personal.

5.2 Conclusion

After analyzing NRK's demographic data and performing classification on their logged-in user base we now arrive at our final conclusions. These conclusions answer the research questions raised in the beginning of this thesis.

Our demographic analysis of the logged-in users suggests that NRK generally reaches age groups 30 and above quite well with respect to gender and geographic location. This is in contrast to the younger group at ages 12-29, which tend to be underrepresented, and therefore reveals where there might be room for improvement in terms of reach.

Predicting demographics is challenging, particularly for the 6-category age classification setting, but more achievable for gender classification and the 4- and binary category age classification cases. This is to the extent that we can outperform a trivial baseline in terms of accuracy, though not by a massive amount.

Our classification study lastly suggests that the accuracy of predictions tend to increase with the quantity of unique contents consumed. The study demonstrated that after a certain number of unique contents viewed, the accuracies could exceed the baseline and improve further with more views. For gender prediction however, the results were better than the baseline results for all quantities of unique contents.

Appendices

APPENDIX A

Appendix A

A.1 Age Group Classification with 6 Classes

		Observed					
		12-17	18-24	25-29	30-49	50-66	67+
Predicted	12-17	91	112	67	264	115	51
	18-24	21	96	62	238	102	26
	25-29	18	82	82	352	118	39
	30-49	15	42	59	606	328	83
	50-66	2	22	23	329	551	174
	67+	3	23	28	274	535	487

Table A.1: Confusion matrix for age classification with 6 age groups, obtained from multinomial regression.

A.2 Age Group Classification with 4 Classes

		Observed			
		12-29	30-49	50-66	67+
Predicted	12-29	595	667	235	98
	30-49	138	712	342	85
	50-66	60	418	633	203
	67+	55	264	537	473

Table A.2: Confusion matrix for age classification with 4 age groups, obtained from multinomial regression.

Ridge Regression

Age group	Recall (SD)	Precision (SD)	F_1 (SD)
12-29	0.705 (0.020)	0.367 (0.010)	0.483 (0.012)
30-49	0.342 (0.010)	0.540 (0.012)	0.419 (0.009)
50-66	0.334 (0.013)	0.485 (0.009)	0.396 (0.010)
67+	0.570 (0.016)	0.356 (0.008)	0.438 (0.010)

Table A.3: Recall, precision, and F_1-score for prediction with 4 age groups using ridge regression.

Lasso Regression

Age group	Recall (SD)	Precision (SD)	F_1 (SD)
12-29	0.702 (0.018)	0.369 (0.010)	0.484 (0.012)
30-49	0.344 (0.009)	0.535 (0.014)	0.419 (0.010)
50-66	0.333 (0.015)	0.482 (0.009)	0.394 (0.012)
67+	0.570 (0.016)	0.358 (0.008)	0.440 (0.010)

Table A.4: Recall, precision, and F_1-score for prediction with 4 age groups using lasso regression.

KNN

Age group	Recall (SD)	Precision (SD)	F_1 (SD)
12-29	0.570 (0.023)	0.294 (0.010)	0.388 (0.013)
30-49	0.306 (0.011)	0.465 (0.013)	0.369 (0.011)
50-66	0.270 (0.007)	0.409 (0.011)	0.325 (0.007)
67+	0.485 (0.017)	0.307 (0.009)	0.376 (0.011)

Table A.5: Recall, precision, and F_1-score for prediction with 4 age groups using KNN.

Random Forest

Age group	Recall (SD)	Precision (SD)	F_1 (SD)
12-29	0.644 (0.011)	0.377 (0.008)	0.475 (0.009)
30-49	0.362 (0.011)	0.525 (0.009)	0.428 (0.010)
50-66	0.357 (0.009)	0.467 (0.007)	0.405 (0.007)
67+	0.541 (0.012)	0.354 (0.010)	0.428 (0.011)

Table A.6: Recall, precision, and F_1-score for prediction with 4 age groups using random forest.

A.3 Binary Age Group Classification

		Observed	
		12-39	40+
Predicted	12-39	1213	1052
	40+	456	2792

Table A.7: Confusion matrix for binary age classification, obtained from random forest.

Logistic Regression			
Age group	Recall (SD)	Precision (SD)	F_1 (SD)
12-39	0.756 (0.006)	0.517 (0.006)	0.614 (0.004)
40+	0.693 (0.009)	0.868 (0.002)	0.771 (0.006)

Table A.8: Recall, precision, and F_1-score for binary age group prediction using logistic regression.

Ridge Regression			
Age group	Recall (SD)	Precision (SD)	F_1 (SD)
12-39	0.755 (0.007)	0.517 (0.007)	0.614 (0.005)
40+	0.693 (0.010)	0.867 (0.003)	0.770 (0.006)

Table A.9: Recall, precision, and F_1-score for binary age group prediction using ridge regression.

Lasso Regression			
Age group	Recall (SD)	Precision (SD)	F_1 (SD)
12-39	0.756 (0.006)	0.516 (0.006)	0.614 (0.004)
40+	0.693 (0.009)	0.867 (0.002)	0.770 (0.005)

Table A.10: Recall, precision, and F_1-score for binary age group prediction using lasso regression.

KNN			
Age group	Recall (SD)	Precision (SD)	F_1 (SD)
12-39	0.714 (0.013)	0.439 (0.007)	0.544 (0.008)
40+	0.604 (0.007)	0.829 (0.007)	0.699 (0.006)

Table A.11: Recall, precision, and F_1-score for binary age group prediction using KNN.

A.4 Gender Classification

		Observed	
		Male	Female
Predicted	Male	2495	558
	Female	921	1539

Table A.12: Confusion matrix for gender classification, obtained from random forest.

Logistic Regression			
Gender	Recall (SD)	Precision (SD)	F_1 (SD)
Male	0.707 (0.007)	0.825 (0.003)	0.761 (0.004)
Female	0.756 (0.006)	0.613 (0.005)	0.677 (0.004)

Table A.13: Recall, precision, and F_1-score for gender prediction using logistic regression.

Ridge Regression			
Gender	Recall (SD)	Precision (SD)	F_1 (SD)
Male	0.703 (0.006)	0.825 (0.003)	0.759 (0.003)
Female	0.756 (0.007)	0.610 (0.004)	0.675 (0.003)

Table A.14: Recall, precision, and F_1-score for gender prediction using ridge regression.

Lasso Regression			
Gender	Recall (SD)	Precision (SD)	F_1 (SD)
Male	0.706 (0.007)	0.825 (0.004)	0.761 (0.004)
Female	0.756 (0.007)	0.612 (0.005)	0.677 (0.004)

Table A.15: Recall, precision, and F_1-score for gender prediction using lasso regression.

KNN			
Gender	Recall (SD)	Precision (SD)	F_1 (SD)
Male	0.616 (0.005)	0.740 (0.005)	0.672 (0.003)
Female	0.647 (0.009)	0.508 (0.004)	0.570 (0.005)

Table A.16: Recall, precision, and F_1-score for gender prediction using KNN.

Bibliography

[1] Albisua, I., Arbelaitz, O., Gurrutxaga, I., Lasarguren, A., Muguerza, J., and Pérez, J. M. "The quest for the optimal class distribution: an approach for enhancing the effectiveness of learning via resampling methods for imbalanced data sets". In: *Progress in Artificial Intelligence* 2.1 (Mar. 2013), pp. 45–63.

[2] Bishop, C. M. *Pattern Recognition and Machine Learning*. Springer, 2006.

[3] Breiman, L. "Random Forests". In: *Machine Learning* 45.1 (Oct. 2001), pp. 5–32.

[4] Breiman, L., Friedman, J. H., Olshen, R. A., and Stone, C. J. *Classification and Regression Trees*. Wadsworth, 1984.

[5] Breiman, L. and Spector, P. "Submodel Selection and Evaluation in Regression. The X-Random Case". In: *International Statistical Review* 60.30 (1992), pp. 291–319.

[6] Carterette, B. *Precision and Recall*. eng. 2009.

[7] Chawla, N. V., Bowyer, K. W., Hall, L. O., and Kegelmyer, W. P. "SMOTE: Synthetic Minority Over-Sampling Technique". In: *Journal of Artificial Intelligence Research* 16 (2002), pp. 321–357.

[8] Chawla, N. V., Japkowicz, N., and Kotcz, A. "Editorial: Special Issue on Learning from Imbalanced Data Sets". In: *SIGKDD Explor. Newsl.* 6.1 (June 2004), pp. 1–6.

[9] Friedman, J., Hastie, T., and Tibshirani, R. "Regularization Paths for Generalized Linear Models via Coordinate Descent". In: *Journal of Statistical Software* 33.1 (2010). glmnet R package version 2.0-16. https://cran.r-project.org/package=glmnet, pp. 1–22.

[10] Hastie, T., Tibshirani, R., and Friedman, J. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. eng. Second Edition. Springer Series in Statistics. New York, NY: Springer New York, 2009.

[11] Hoerl, A. and Kennard, R. "Ridge Regression: Biased Estimation for Nonorthogonal Problems". In: *Technometrics* 12.1 (1970), pp. 55–67.

[12] Hu, J., Zeng, H.-J., Li, H., Niu, C., and Chen, Z. "Demographic Prediction Based on User's Browsing Behavior". In: *Proceedings of the 16th International Conference on World Wide Web*. WWW '07. Banff, Alberta, Canada: ACM, 2007, pp. 151–160.

[13] Hu, Y., Koren, Y., and Volinsky, C. "Collaborative Filtering for Implicit Feedback Datasets". eng. In: *2008 Eighth IEEE International Conference on Data Mining*. IEEE, 2008, pp. 263–272.

[14] Japkowicz, N. and Stephen, S. "The class imbalance problem: A systematic study". In: *Intelligent Data Analysis* 6.5 (2002), pp. 429–449.

[15] Kenny, P. *Decisions from Data: Statistical Analysis for Professional Success*. Apress, 2014.

[16] Kohavi, R. "A Study of Cross-validation and Bootstrap for Accuracy Estimation and Model Selection". In: *Proceedings of the 14th International Joint Conference on Artificial Intelligence - Volume 2*. IJCAI'95. Montreal, Quebec, Canada: Morgan Kaufmann Publishers Inc., 1995, pp. 1137–1143.

[17] Kosinski, M., Stillwell, D., and Graepel, T. "Private traits and attributes are predictable from digital records of human behavior". In: *Proceedings of the National Academy of Sciences* 110.15 (2013), pp. 5802–5805.

[18] Kouiroukidis, N. and Evangelidis, G. "The Effects of Dimensionality Curse in High Dimensional kNN Search". eng. In: *2011 15th Panhellenic Conference on Informatics*. IEEE, 2011, pp. 41–45.

[19] Krismayer, T., Schedl, M., Knees, P., and Rabiser, R. "Predicting user demographics from music listening information". In: *Multimedia Tools and Applications* 78.3 (Feb. 2019), pp. 2897–2920.

[20] Krismayer, T., Schedl, M., Knees, P., and Rabiser, R. "Prediction of User Demographics from Music Listening Habits". In: *Proceedings of the 15th International Workshop on Content-Based Multimedia Indexing*. CBMI '17. Florence, Italy: ACM, 2017, 8:1–8:7.

[21] Kuhn, M. "Building Predictive Models in R Using the caret Package". In: *Journal of Statistical Software, Articles* 28.5 (2008). caret R package version 6.0-81. https://cran.r-project.org/package=caret, pp. 1–26.

[22] Li, C., Wan, J., and Wang, B. "Personality Prediction of Social Network Users". In: *2017 16th International Symposium on Distributed Computing and Applications to Business, Engineering and Science (DCABES)*. Oct. 2017, pp. 84–87.

[23] Liaw, A. and Wiener, M. "Classification and Regression by randomForest". In: *R News* 2.3 (2002). randomForest R package version 4.6-14. https://cran.r-project.org/package=randomForest, pp. 18–22.

[24] Lopez, V., Fernandez, A., Garcia, S., Palade, V., and Herrer, F. "An insight into classification with imbalanced data: Empirical results and current trends on using data intrinsic characteristics". In: *Information Sciences* 250 (2013), pp. 113–141.

[25] Moepya, S. O., Akhoury, S. S., and Nelwamondo, F. V. "Applying Cost-Sensitive Classification for Financial Fraud Detection under High Class-Imbalance". In: *2014 IEEE International Conference on Data Mining Workshop* (Dec. 2014), pp. 183–192.

[26] Posten Norge AS. *Postnummer i Norge*. Accessed March 2018. URL: https://data.norge.no/data/posten-norge/postnummer-i-norge.

[27] R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing. Vienna, Austria, 2018.

[28] Solinger, C., Hirshfield, L., Hirshfield, S., Friendman, R., and Leper, C. "Beyond Facebook Personality Prediction:" in: *Social Computing and Social Media*. Ed. by Meiselwitz, G. Vol. 8531. Cham: Springer International Publishing, 2014, pp. 486–493.

[29] Statistisk Sentralbyrå. *Befolkning*. Tabell 11727 via SSB API. Accessed March 2018. 2017. URL: https://www.ssb.no/befolkning/statistikker/folkemengde/aar-per-1-januar#scroll.

[30] Statistisk Sentralbyrå. *Tettsteders befolkning og areal*. Accessed March 2018. 2017. URL: https://www.ssb.no/beftett.

[31] "Strengthening outcomes of retailer–consumer relationships: The dual impact of relationship marketing tactics and consumer personality". In: *Journal of Business Research* 56.3 (2003), pp. 177–190.

[32] Tibshirani, R. "Regression Shrinkage and Selection via the Lasso". In: *Journal of the Royal Statistical Society. Series B (Methodological)* 58.1 (1996), pp. 267–288.

[33] Van Rijsbergen, C. *Information retrieval*. London, 1979.

[34] Venables, W. N. and Ripley, B. D. *Modern Applied Statistics with S*. Fourth. nnet R package version 7.3-12. https://CRAN.R-project.org/package=nnet. New York: Springer, 2002.

[35] Yadav, S. and Shukla, S. "Analysis of k-Fold Cross-Validation over Hold-Out Validation on Colossal Datasets for Quality Classification". eng. In: *2016 IEEE 6th International Conference on Advanced Computing (IACC)*. IEEE, 2016, pp. 78–83.

Program and Packages

[9] Friedman, J., Hastie, T., and Tibshirani, R. "Regularization Paths for Generalized Linear Models via Coordinate Descent". In: *Journal of Statistical Software* 33.1 (2010). glmnet R package version 2.0-16. https://cran.r-project.org/package=glmnet, pp. 1–22.

[21] Kuhn, M. "Building Predictive Models in R Using the caret Package". In: *Journal of Statistical Software, Articles* 28.5 (2008). caret R package version 6.0-81. https://cran.r-project.org/package=caret, pp. 1–26.

[23] Liaw, A. and Wiener, M. "Classification and Regression by randomForest". In: *R News* 2.3 (2002). randomForest R package version 4.6-14. https://cran.r-project.org/package=randomForest, pp. 18–22.

[27] R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing. Vienna, Austria, 2018.

[34] Venables, W. N. and Ripley, B. D. *Modern Applied Statistics with S*. Fourth. nnet R package version 7.3-12. https://CRAN.R-project.org/package=nnet. New York: Springer, 2002.