Norsk Regnesentral
NORWEGIAN COMPUTING CENTER

Credit Scoring using Deep Learning and how to explain predictions from black-box models

Cramérsällskapets årsmöte ,
Stockholm, September 15th, 2020

Kjersti Aas
Assistant research director
Norsk Regnesentral

Norsk Regnesentral

- Private foundation
- Applied contracted research
- More than 40 years experience in statistics and machine learning
- 75 MNOK turnover in statistics / machine learning
- 100 clients annually
- Finance and insurance is one of our largest market areas

Norsk Regnesentral
NORWEGIAN COMPUTING CENTER

Credit Scoring using Deep Learning

Problem

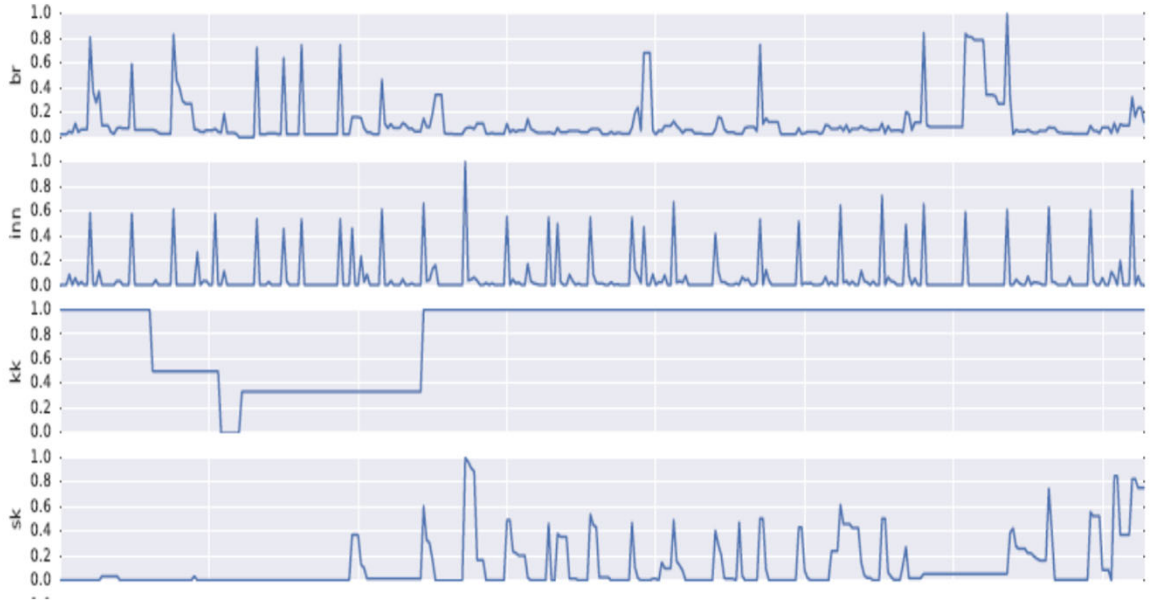
- Want to predict the probability of mortgage default*.
- The current state of art method is a logistic regression model with handcrafted features.
- The typical variables used are number of outstanding accounts, delinquent accounts, monthly income and demographic data, such as age and marital status.
- The most important variable is **the number of previous overdue payments.**

*Default = Bill past due for more than 90 days

Transaction data

- The information about overdue payments is only available for customers who already have been granted a mortgage.
- We wanted to investigate whether it is possible to predict the probability of default (PD) earlier, i.e. at the time of loan application.
- In Norway, debit cards are by far the most common payment form - electronic credit transfers account for nearly 90% of all payments.
- Hence, transactional data may provide a useful description of user behavior and consumer credit risk.

Example



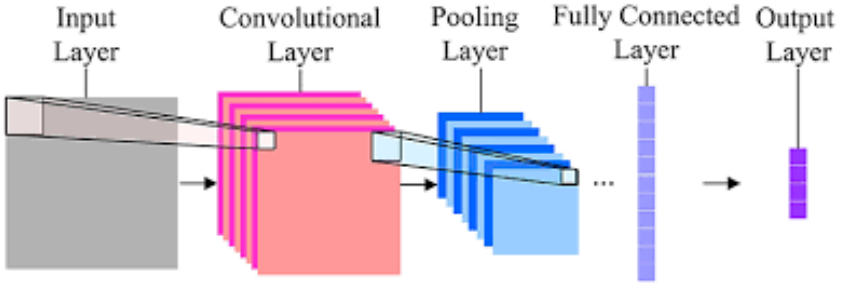
The transaction information consists of:

- The daily balance on the consumers checking account
- The daily balance on the consumers savings account
- The daily balance on the consumers credit card account
- The daily number of transactions on the checking account
- The daily amount into the checking account.

Convolutional Neural Network (CNN)

- Our approach is to view the PD prediction problem as a time series classification problem.

- To classify the time series we use deep learning, or more specifically a Convolutional Neural Network (CNN).
- A CNN is a neural network with different types of hidden layers:
  - Convolutional layers
  - Max pooling layers
  - Fully-connected layers



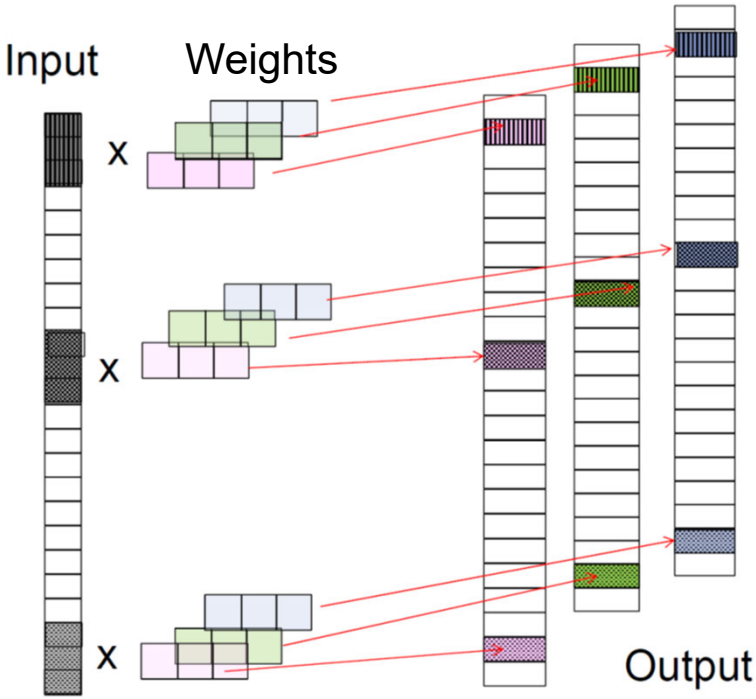
Convolutional layer

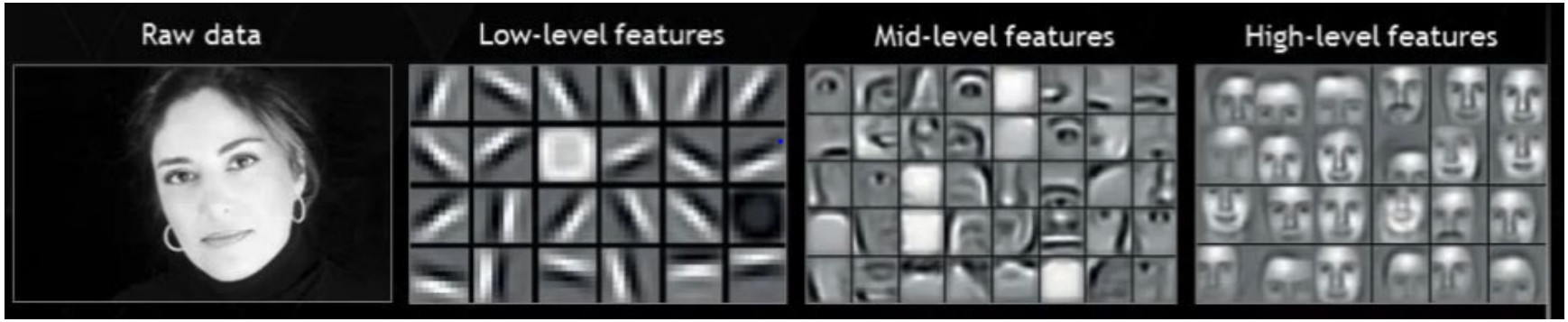
- A convolutional layer consists of J filters of size $2s+1$
- The j'th filter produces the output $y_{t,j}$ given by:

$$y_{t,j} = f_a\left(\sum_{i=-s}^{s} w_{i,j}\, x_{t+i} + b_j\right); \qquad j = 1, \ldots, J$$

- The most common activation function is

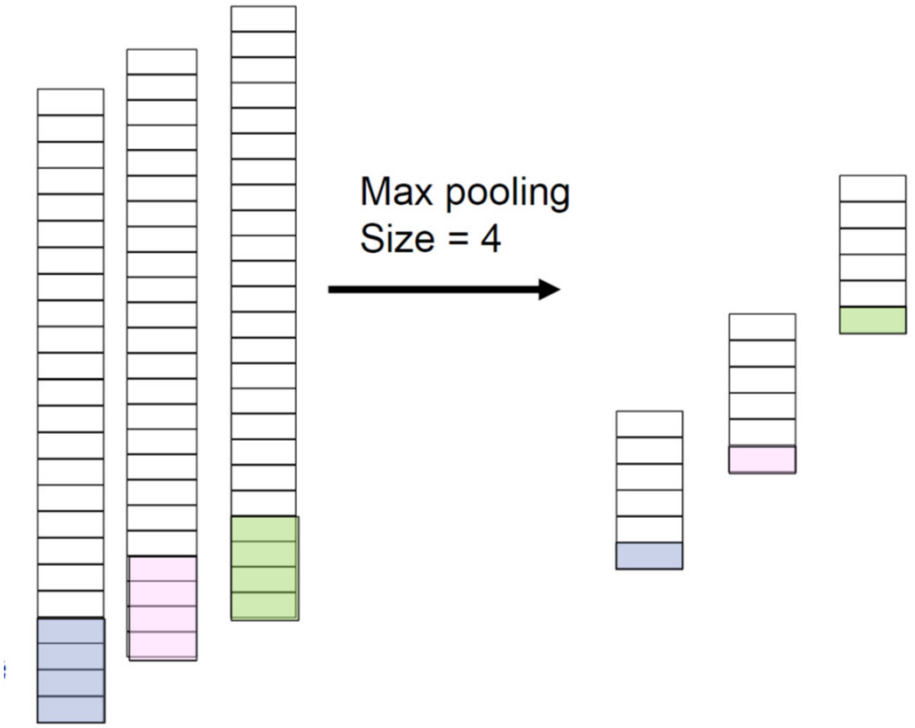
$$f_a(x) = max(x, 0)$$





Max-pooling layer

- Pooling layers reduce the dimensions of the data by combining groups of outputs from one layer into a single neuron in the next layer.
- Max pooling uses the maximum value of each group.

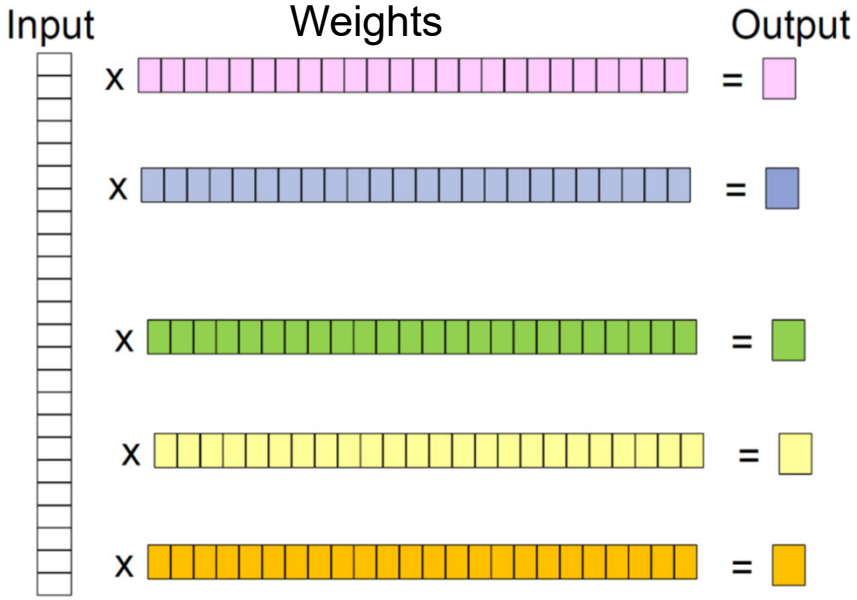


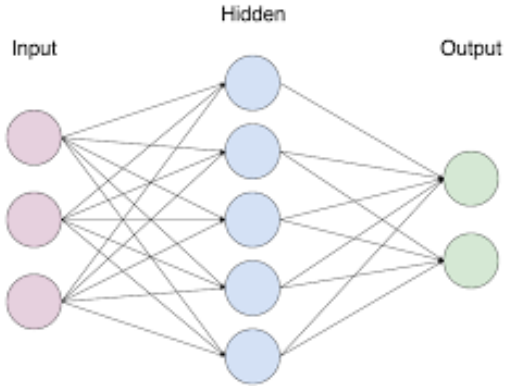
Fully connected layer

- Fully connected layers connect every neuron in one layer to every neuron in another layer.
- Output node j has the value:

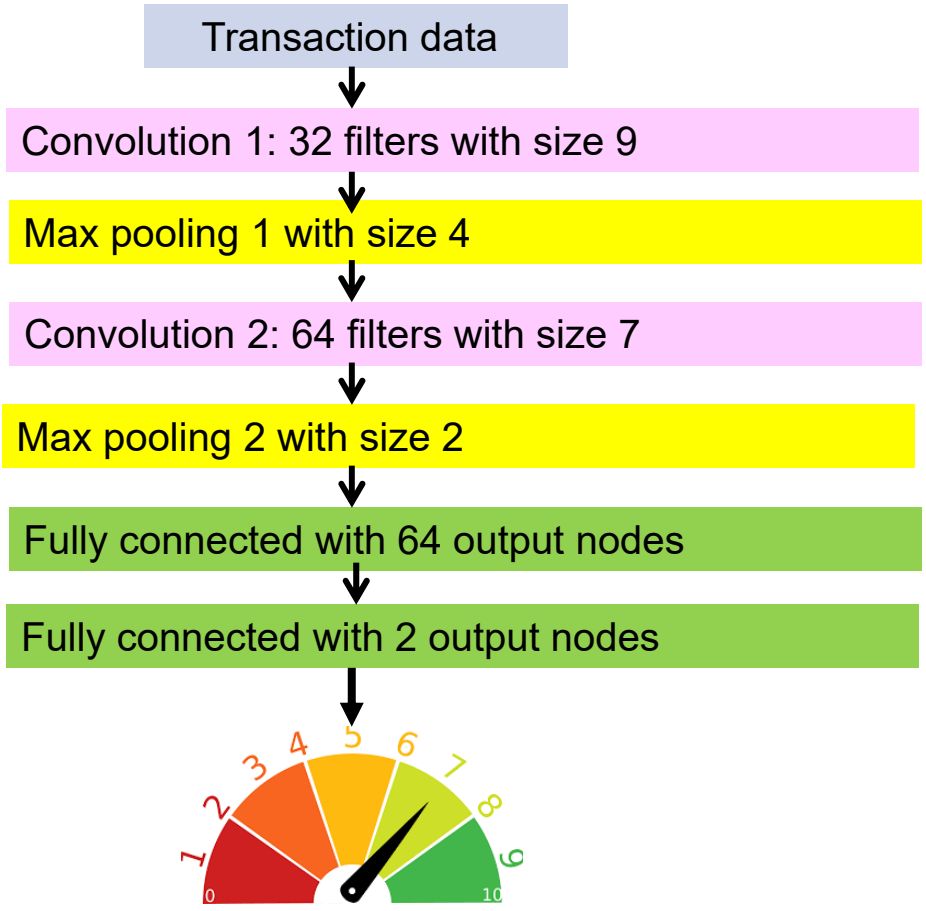
$$y_j = f\left(\sum_{i=1}^{N} w_{i,j}\, x_i + b_j\right)$$

- It is in principle the same as the traditional MLP neural network.





Our CNN:



Probability of default

199 235 parameters!

A very complex model will fit your historical data well, but it will have low predictive power!

You always need a validation data set!

Both for training and validation you need to know the truth

Use ReLU in all layers except for the last, where softmax is used.

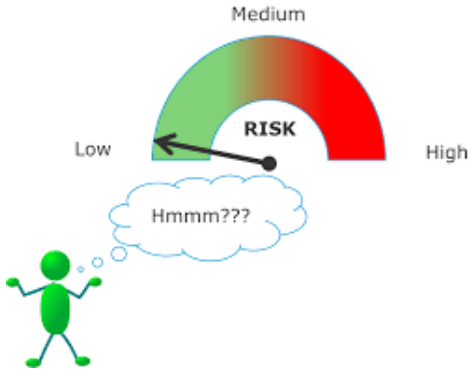
Data set

- 20,989 mortgage customers
- **Training set:**
  - Transaction data from the period 31.12.2011 – 31.12.2013
  - Default/non-default during the period 01.01.14 – 01.01.15
- **Validation set:**
  - A random subset from the training set
- **Test set:**
  - Transaction data from 28.02.2014 – 28.02.15
  - Default/non-default during the period 01.03.15 – 01.03.16

Use data augmentation to increase the data set: Many one-year transaction periods for each customer with the same default period.

Very positive results

- Better identification of low risk group:
  - Increased from 80% with existing model to 95% with the new model.
- Good identification of high risk group:
  - 50% of those who actually defaulted was among the 1% with highest risk according to the new model.



https://blogs.dnvgl.com/software/2016/02/now-time-rethink-concept-low-risk-facility-really-exist/

The value for DNB

Increased digitalization
Less credit losses
Decreased capital requirements
Identify more profitable customers

Manual resources more focused on the complex cases

Expert Systems With Applications 102 (2018) 207–217

Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa

Predicting mortgage default using convolutional neural networks

Håvard Kvamme[a,*], Nikolai Sellereite[b], Kjersti Aas[b], Steffen Sjursen[c]

[a] Department of Mathematics, University of Oslo, Niels Henrik Abels hus Moltke Moes vei 35, Oslo 0851, Norway
[b] Statistical Analysis, Machine Learning and Image Analysis, Norwegian Computing Center, Gaustadalleen 23a, Oslo 0373, Norway
[c] Group Risk Modelling, DNB ASA, Dronning Eufemias gate 30, Oslo 0191, Norway

ARTICLE INFO

Article history:
Received 15 August 2017
Revised 17 February 2018
Accepted 18 February 2018
Available online 19 February 2018

Keywords:
Consumer credit risk
Machine learning
Deep learning
Mortgage default model
Time series

ABSTRACT

We predict mortgage default by applying convolutional neural networks to consumer transaction data. For each consumer we have the balances of the checking account, savings account, and the credit card, in addition to the daily number of transactions on the checking account, and amount transferred into the checking account. With no other information about each consumer we are able to achieve a ROC AUC of 0.918 for the networks, and 0.926 for the networks in combination with a random forests classifier.

1. Introduction

The ability to discriminate bad customers from good ones is important for banks and other lending companies. A small improvement in prediction accuracy may result in a large gain in profitability. Early identification of high risk consumers may aid the prevention of loan defaults and help the consumers to better manage their personal economy.

In credit scoring, one builds a model for the correspondence between default and various loan obligor characteristics based on a relevant sample of people, and use this model to predict the probability that a person will repay his debts.

There is an extensive literature on credit scoring, both for assessing private loans (Butaru et al., 2016; Chi & Hsu, 2012; Khandani, Kim, & Lo, 2010; Sousa, Gama, & Brandão, 2016) and corporate loans (Jones, Johnstone, & Wilson, 2015; Ravi Kumar & Ravi, 2007). Some recent work include Abellán and Castellano (2017); Chen, Zhou, Wang, and Li (2017); Xia, Liu, Li, and Liu (2017), and Barboza, Kimura, and Altman (2017). For an overview and comparison of papers, see García, Marqués, and Sánchez (2014) and Lessmann, Baesens, Seow, and Thomas (2015).

All the papers above attempt to model delinquencies and defaults by applying machine learning algorithms to a set of exthat is available to researchers (see e.g. Lessmann et al., 2015), the constructed explanatory variables tend to be quite similar. Papers typically use information from credit bureaus, such as number of outstanding accounts, delinquent accounts, and balance on other loans; individual account characteristics, such as current balance of the individuals accounts and monthly income; and demographic data, such as age and marital status. Butaru et al. (2016) also include macroeconomic variables, such as interest rates and unemployment statistics, as an attempt to make the delinquency model generalize better over longer periods of time.

As all these papers use similar explanatory variables, the researchers commonly explore differences between scoring models rather than the benefit of adding new explanatory variables. There are however some exceptions: Khandani et al. (2010) explore the benefit of adding information from detailed purchase volumes to their models. This includes travel expenses, gas station expenses, bar expenses, etc. Chi and Hsu (2012) also introduce consumer transaction data through an aggregated measure called average utilization ratio of credit.

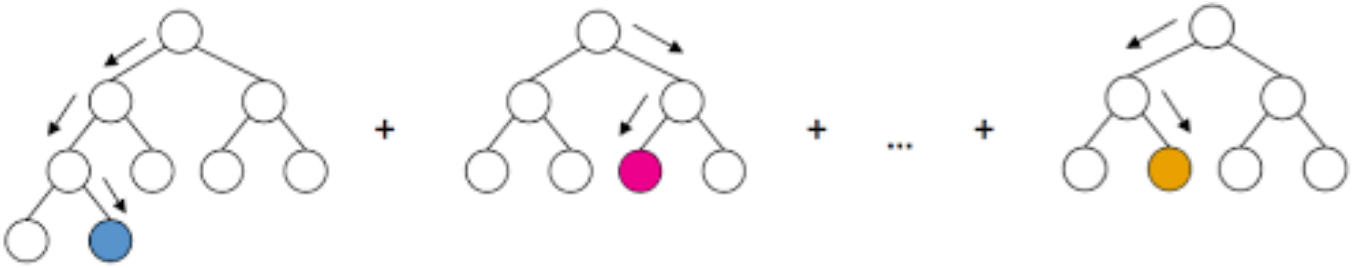
In this paper we further investigate how transaction data can be used for credit scoring. In a joint research with Norway's largest financial service group, DNB, we use transaction data to predict mortgage defaults. In 2012, the average Norwegian made 323 card

Norsk Regnesentral
NORWEGIAN COMPUTING CENTER

Explaining predictions from black-box models

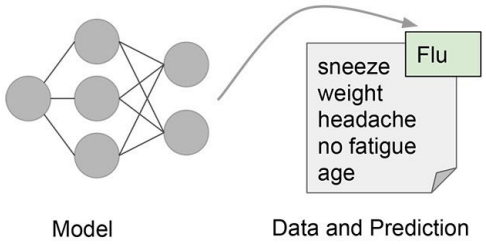
Example: Mortgage robot

- XGBoost model which predicts mortgage default
- 28 covariates extracted from 6 transaction time series
  - **Example 1:** Mean value of the daily balance on the consumers checking account during the last 365 days.
  - **Example 2:** Standard deviation of the daily balance on the consumers savings account during the last 365 days.
- Why was Ola Nordmann rejected a loan?



Difficult problem

- To trust a model you need to know how it works!
- Which input variables are most important?
  - Global explanations
  - Local explanations
- Difficult problem!
  - Not even for the simple linear regression model it is straightforward to determine the importance of each variable if the variables are not independent!

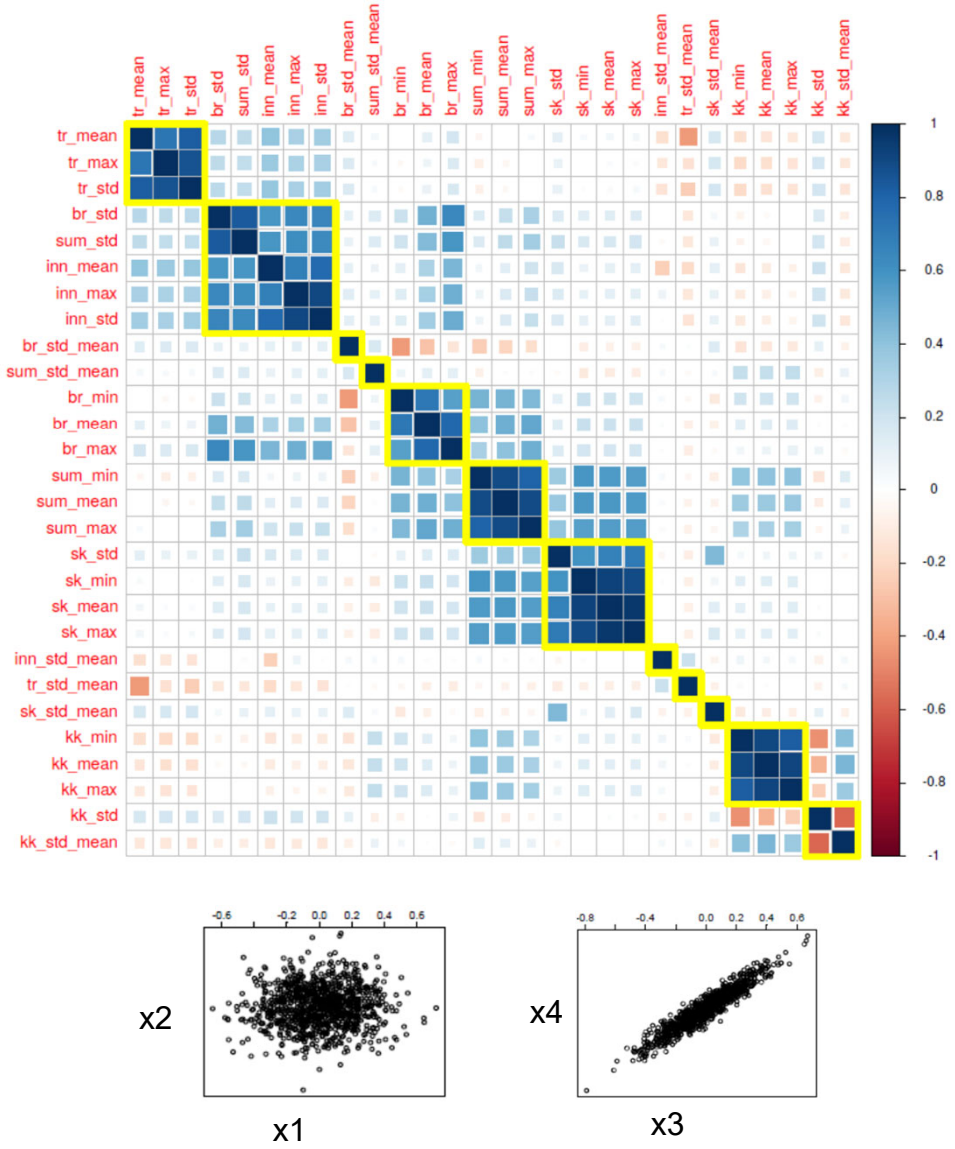


$$y = 0.5\,x_1 + 0.2\,x_2 + 0.3\,x_3 + \epsilon$$

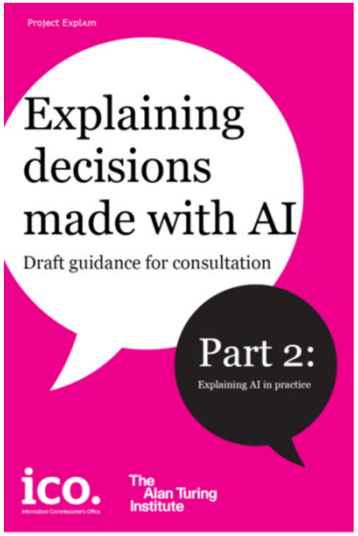
What is the global importance of x_1, x_2 and x_3?

Dependence

- Usually data sets used to estimate machine learning models have dependent variables.
- Throwing out variables that are highly correlated with other variables often reduces the model performance.



Local explanation methods



- Model-specific methods:
  - Deep Lift: For deep learning models
  - TreeSHAP: For XGBoost models
- Model-agnostic methods:
  - LIME Local linear regression
  - Shapley Based on concepts from game theory
  - Counterfactual explanations: Which variables should be altered to obtain a different decision?

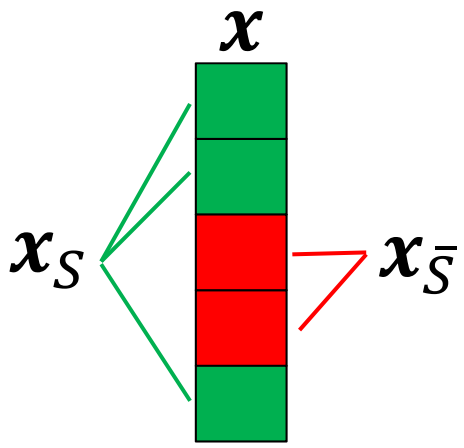
Shapley values

- Based on concepts from game theory.
- Idea: Predictions can be explained by assuming that each variable is a player in a game where the prediction is the payout.
- The difference between the prediction and the average prediction is fairly distributed among the variables.
- Gives an explicit formula for the importance of every variable.

$$\phi_j = \sum_{S \subseteq M \setminus \{j\}} w(S) \left(v(S \cup \{j\}) - v(S) \right)$$

Shapley values for prediction explanation

- Players = covariates $(x_1, \ldots, x_p)$
- The instance to be explained = $\boldsymbol{x}^*$
- Payoff = prediction $(f(\boldsymbol{x}^*))$
- Contribution function: $v(S) = E[f(\boldsymbol{x})|\boldsymbol{x}_S = \boldsymbol{x}_S^*]$
- Properties



$$f(\boldsymbol{x}^*) = \sum_{j=0}^{p} \phi_j \qquad \phi_0 = E[f(\boldsymbol{x})]$$

f indep. of $x_j \Rightarrow \phi_j = 0$, $\quad x_i, x_j$ same contribution $\Rightarrow \phi_i = \phi_j$

The Shapley value is the average expected marginal contribution of one player after all possible combinations have been considered.

Challenges

Two main challenges:

- The computational complexity of the Shapley formula
  - Partly solved by subset sampling (KernelSHAP method)
- Estimating the contribution function
  - Not trivial if the model is non-linear and the covariates are dependent
  - Previous methods assume independent covariates

Our contribution

- We take the dependence between the features into account when estimating the contribution function.
- The contribution value may be computed as follows:

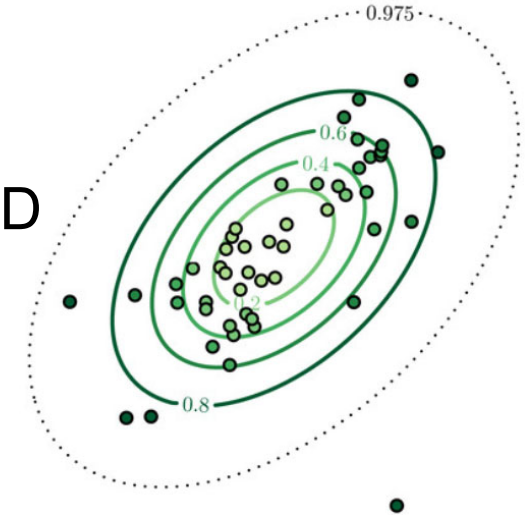
$$E[f(\boldsymbol{x})|\boldsymbol{x}_S = \boldsymbol{x}_S^*] = E[f(\boldsymbol{x}_{\bar{S}}, \boldsymbol{x}_S)|\boldsymbol{x}_S = \boldsymbol{x}_S^*]$$
$$= \int f(\boldsymbol{x}_{\bar{S}}, \boldsymbol{x}_S^*)\, p(\boldsymbol{x}_{\bar{S}}|\boldsymbol{x}_S = \boldsymbol{x}_S^*) d\boldsymbol{x}_{\bar{S}}.$$

- We use Monte Carlo integration to compute the integral.
- Hence, we need to be able to generate samples from the conditional distribution $p(\boldsymbol{x}_{\bar{S}}|\boldsymbol{x}_S = \boldsymbol{x}_S^*)$ where $\boldsymbol{x}_{\bar{S}}$ is the part of $\boldsymbol{x}$ not in $\boldsymbol{x}_S$

Continuous variables

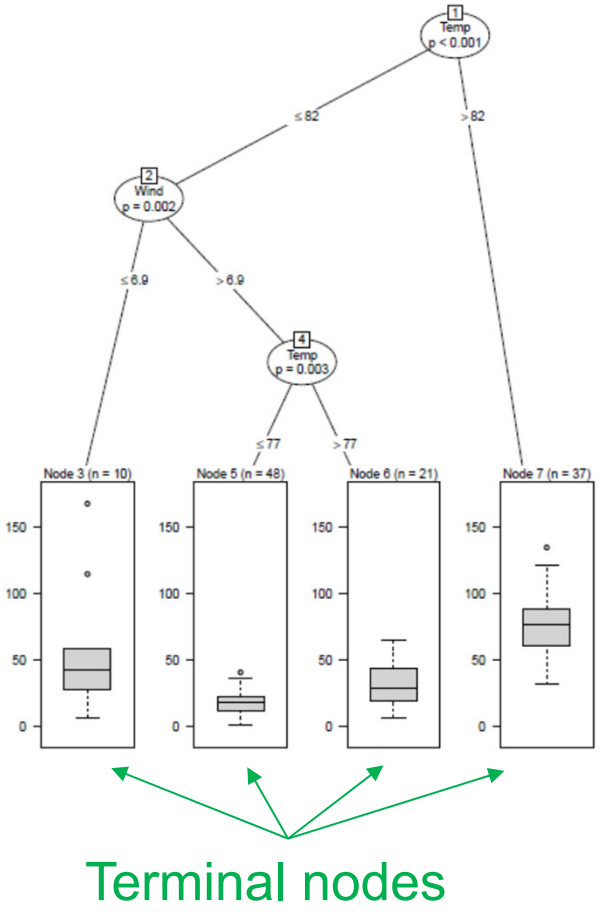
- We propose 3 approaches for estimating $p(\boldsymbol{x}_{\bar{S}}|\boldsymbol{x}_S = \boldsymbol{x}_S^*)$:

1. Assume $p(\boldsymbol{x})$ Gaussian => analytical $p(\boldsymbol{x}_{\bar{S}}|\boldsymbol{x}_S = \boldsymbol{x}_S^*)$
2. Use an empirical (conditional) approach where training observations at $\boldsymbol{x}_{\bar{S}}^k$ are weighted by proximity of $\boldsymbol{x}_S^k$ to $\boldsymbol{x}_S^*$
3. Use a combination of the two approaches
   - Use the empirical approach when $|\boldsymbol{x}_S|$ < D
   - Use the Gaussian approach otherwise



Categorical and mixed variables

- Fit a multivariate regression tree with response $\boldsymbol{x}_{\bar{S}}$ and covariates $\boldsymbol{x}_S$ using the training data.
- Determine the terminal node in this tree to which $\boldsymbol{x}_S^*$ belongs.
- Approximate $p(\boldsymbol{x}_{\bar{S}}|\boldsymbol{x}_S = \boldsymbol{x}_S^*)$ by sampling K times from the training observations that also attained this node number.



Evaluation

- No ground truth ⇒ not obvious how to evaluate the different approaches.
- We have compared Shapley with and without taking dependence into account in several controlled experiments.
  - Linear and non-linear models
  - Gaussian and non-Gaussian distributions

- Our results show that the combined approach is superior when the model is non-linear and the data follows a non-linear distribution.

Want to know more?

Read our papers on arXiv
arxiv.org/abs/1903.10464
arxiv.org/abs/2007.01027

Check out our R-package
shapr on Github and CRAN
github.com/NorskRegnesentral/shapr
https://cran.r-project.org/web/packages/shapr/index.html

Norsk Regnesentral
NORWEGIAN COMPUTING CENTER

Thank you
for your attention