Opening the black box

Individual prediction explanation

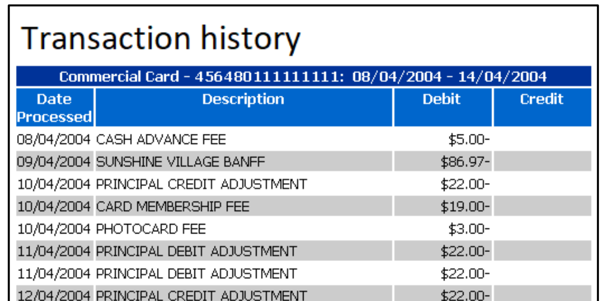
Martin Jullum

Big Insight Day 2019
Oslo, November 14th 2019

The rest of the team at NR




Example: Bank creates mortgage robot



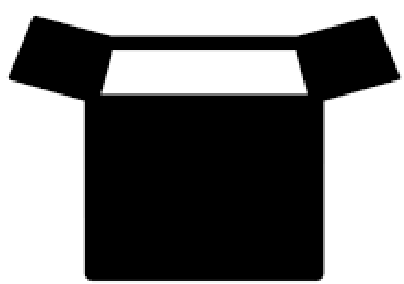
Transaction history

Commercial Card - 456480111111111: 08/04/2004 - 14/04/2004			
Date Processed	Description	Debit	Credit
08/04/2004	CASH ADVANCE FEE	$5.00-	
09/04/2004	SUNSHINE VILLAGE BANFF	$86.97-	
10/04/2004	PRINCIPAL CREDIT ADJUSTMENT	$22.00-	
10/04/2004	CARD MEMBERSHIP FEE	$19.00-	
10/04/2004	PHOTOCARD FEE	$3.00-	
11/04/2004	PRINCIPAL DEBIT ADJUSTMENT	$22.00-	
11/04/2004	PRINCIPAL DEBIT ADJUSTMENT	$22.00-	
12/04/2004	PRINCIPAL CREDIT ADJUSTMENT	$22.00-	

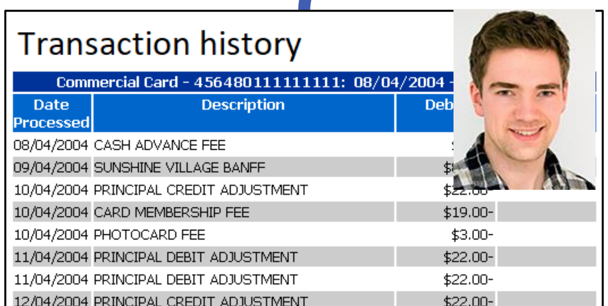
Age/gender/prev. loans…

&

Defaulted loan?



Example: Bank creates mortgage robot



Transaction history

Commercial Card - 4564801111111111: 08/04/2004 -		
Date Processed	Description	Deb
08/04/2004	CASH ADVANCE FEE	
09/04/2004	SUNSHINE VILLAGE BANFF	
10/04/2004	PRINCIPAL CREDIT ADJUSTMENT	
10/04/2004	CARD MEMBERSHIP FEE	$19.00-
10/04/2004	PHOTOCARD FEE	$3.00-
11/04/2004	PRINCIPAL DEBIT ADJUSTMENT	$22.00-
11/04/2004	PRINCIPAL DEBIT ADJUSTMENT	$22.00-
12/04/2004	PRINCIPAL CREDIT ADJUSTMENT	$22.00-

Age/gender/prev. loans…

P(Mortgage default)

Mortgage granted ?

$$\boldsymbol{x} \longrightarrow f(\boldsymbol{x}) \longrightarrow p = 0.7$$

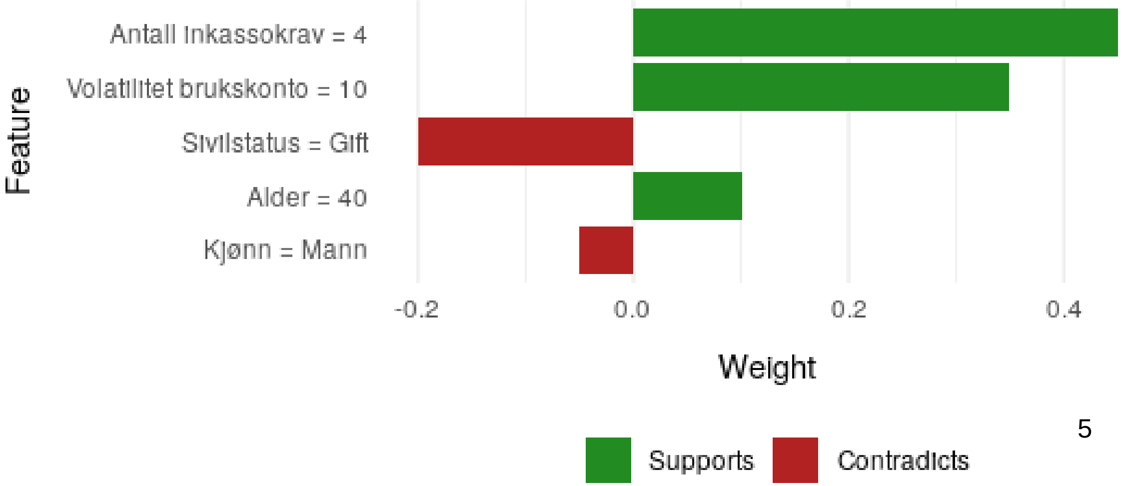
<u>Why</u> was  rejected a loan?

Individual prediction explanation

- **NOT** a general explanation of the black-box model
- $x = x^*$: Transaction history/covariates for 

Explanation for $f(x^*)$ = 70%

- Which covariates "contributed the most" to increase/decrease the prediction to exactly $f(x^*)$= 70%?



Why is this important?

- Customers may have a “right to an explanation”
- Also builds trust to the “robot”

Prediction explanation in general

- Assume we have trained a statistical or machine learning model to describe a response variable Y based on a set of covariates $\boldsymbol{x} = (x_1, \dots, x_p), i.e$:

$$Y \approx f(\boldsymbol{x})$$

- f applied to predict Y for a new set of covariates $\boldsymbol{x} = \boldsymbol{x}^*$

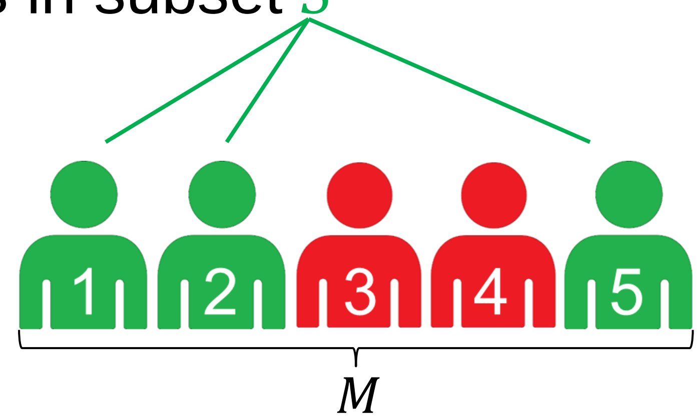
- Want explain the prediction by translating $f(\boldsymbol{x}^*)$ to scores $\phi_1, \dots, \phi_p$ representing the contribution of the covariates $\boldsymbol{x}^*$

Shapley values

- Concept from (cooperative) game theory in the 1950s
- Used to distribute the total payoff to the players
- Explicit formula for the "fair" payment to every player j:

$$\phi_j = \sum_{S \subseteq M \setminus \{j\}} w(S)\left(v(S \cup \{j\}) - v(S)\right), \qquad w(S) \text{ is a weight function}$$

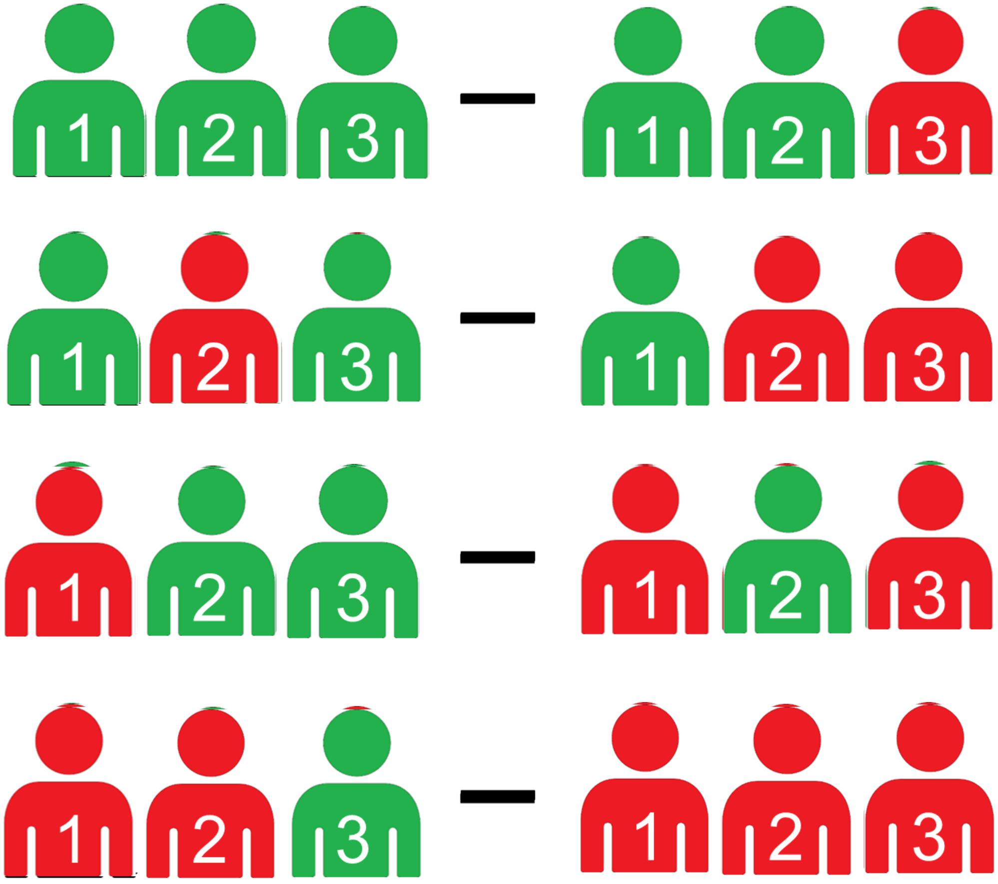
$v(S)$ is the payoff with only players in subset S



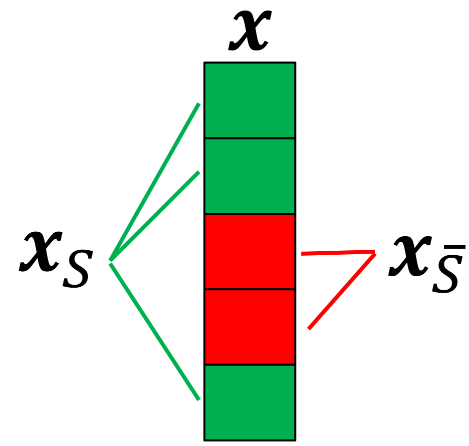
- Several mathematical optimality properties

Intuition behind the Shapley formula

Game with 3 players 



Shapley values for prediction explanation



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

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

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

- Rough interpretation of ϕ_j: **How does the prediction change when you don't know the value of** x_j

Shapley values for prediction explanation

- 2 main challenges

1. The computational complexity in the Shapley formula

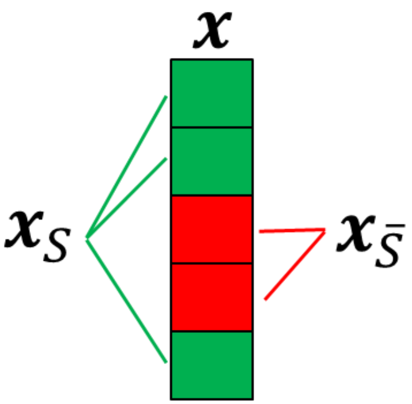
$$\phi_j = \sum_{S \subseteq M \setminus \{j\}} w(S)\left(v(S \cup \{j\}) - v(S)\right)$$

   - Partly solved by cleverly reducing the sum by subset sampling (KernelSHAP; Lundberg & Lee, 2017)

Shapley values for prediction explanation

- 2 main challenges

Recall



2. Estimating the contribution function

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

- Previous methods
  - Approximates $v(S) \approx \int f(\boldsymbol{x}_{\bar{S}}, \boldsymbol{x}_S^*) p(\boldsymbol{x}_{\bar{S}}) \mathrm{d}\boldsymbol{x}_{\bar{S}}$,
  - Estimates $p(\boldsymbol{x}_{\bar{S}})$ using the empirical distribution of the training data
  - Monte Carlo integration to solve the integral

  This assumes covariates are independent!

Consequences of the independence assumption

- Requires evaluating $f(\boldsymbol{x}_{\bar{S}}, \boldsymbol{x}_S)$ at potentially **unlikely or illegal** combinations of $\boldsymbol{x}_{\bar{S}}$ and $\boldsymbol{x}_S$

- Example 1
  - Number of transactions to Switzerland: 0
  - Average transaction amount to Switzerland: 1000 NOK

- Example 2
  - Age: 17
  - Marital status: Widow
  - Profession: Professor

Our idea

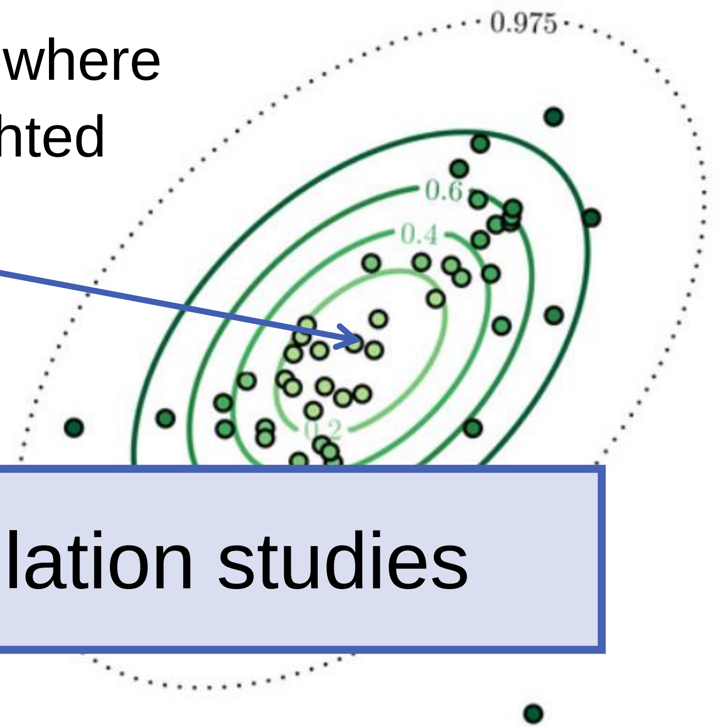
Estimate $p(\boldsymbol{x}_{\bar{S}} | \boldsymbol{x}_S = \boldsymbol{x}_S^*)$ properly

+

Monte Carlo integration

Our contribution: continuous variables

- 3 approaches for estimating $p(\boldsymbol{x}_{\bar{S}}|\boldsymbol{x}_S = \boldsymbol{x}_S^*)$
  - Assume $p(\boldsymbol{x})$ Gaussian => analytical $p(\boldsymbol{x}_{\bar{S}}|\boldsymbol{x}_S = \boldsymbol{x}_S^*)$
  - Assume Gaussian copula => transformation + analytical expression
  - An empirical (conditional) approach where training observations at $\boldsymbol{x}_{\bar{S}}^k$ are weighted by proximity of $\boldsymbol{x}_S^k$ to $\boldsymbol{x}_S^*$



<u>BIG</u> improvements in simulation studies

Explaining sick leave predictions

- NAV is modelling how long individuals are on sick leave
  - Used by case workers to schedule follow-up meetings
- Case workers need to understand the “reasoning” of the individual predictions
- Modelling based on
  - age, gender, sick leave history, type of business etc.
  - Several categorical variables with many levels
- Need methodology for prediction explanation which can handle categorical variables

Our idea

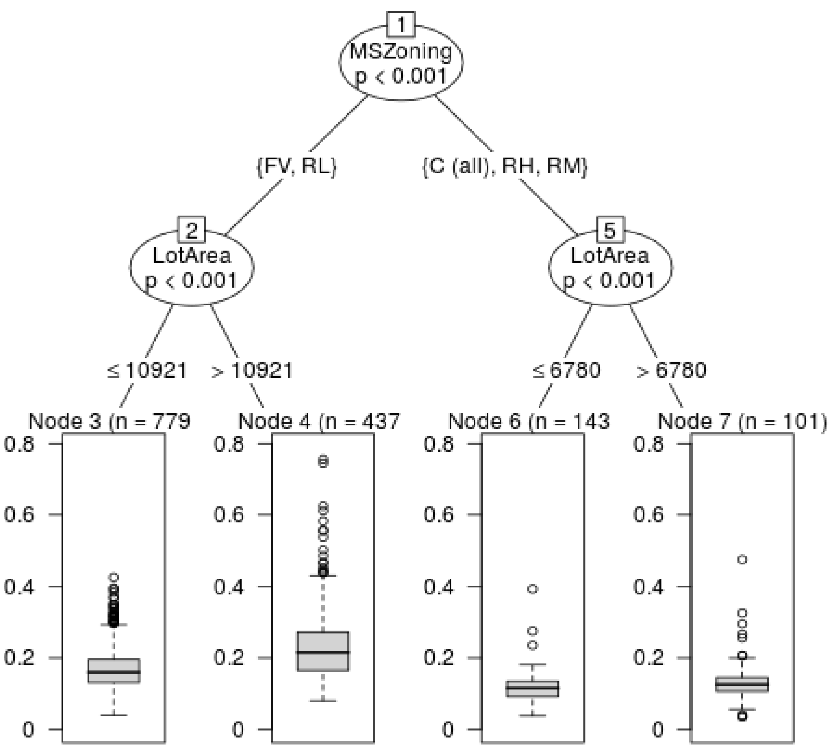
Estimate $p(\boldsymbol{x}_{\bar{S}} | \boldsymbol{x}_S = \boldsymbol{x}_S^*)$ properly

+

Monte Carlo integration

Our contribution: categorical/mixed variables

- Estimating $p(\boldsymbol{x}_{\bar{S}}|\boldsymbol{x}_S = \boldsymbol{x}_S^*)$
  - For every subset S, fit a **(multivariate) regression tree** to $\boldsymbol{y} = \boldsymbol{x}_{\bar{S}}$ based on $\boldsymbol{x}_S$ using the training data
  - Approximate $p(\boldsymbol{x}_{\bar{S}}|\boldsymbol{x}_S = \boldsymbol{x}_S^*)$ by the empirical distribution of the training observations ($\boldsymbol{x}_{\bar{S}}$) within the terminal node of $\boldsymbol{x}_S = \boldsymbol{x}_S^*$



Want to know more?

Read our paper on arXiv
arxiv.org/abs/1903.10464

Check out our R-package
shapr on Github
github.com/NorskRegnesentral/shapr

Talk to any of us