De (on)mogelijkheden van big data in de gezondeidszorg

ABC van machine learning

Derek de Beurs, PhD

NIVEL
Kennis voor betere zorg

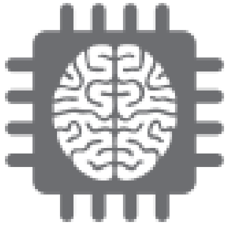
VU

Take home

- Big data en machine learning in de gezondheidszorg blijven
- Gezondheidszorg heeft andere uitdagingen dan het herkennen van een Cihuahua
- Verdiep je als behandelaar/onderzoeker ook in machine learning

Artificial Intelligence in Healthcare is here to stay

It's no longer a question of if, but how fast

Last decade	Current decade	Next decade
Medical Products	**Medical Platforms**	**Medical Solutions**
Equipment, Hardware, Consumables	Wearable, Big Data, Health Analytics	Robotics, AI, Augmented Reality
		
Differentiation is solely through product innovation. Focused on historic and evidence based-care.	Differentiation by providing services to key stakeholders. Focused on real time outcome based-care.	Differentiation via intelligent solutions for evidence/outcome basec health. Focused on preventive care.

Source: Frost & Sullivan, 'Transforming healthcare through artificial intelligence systems', 2016

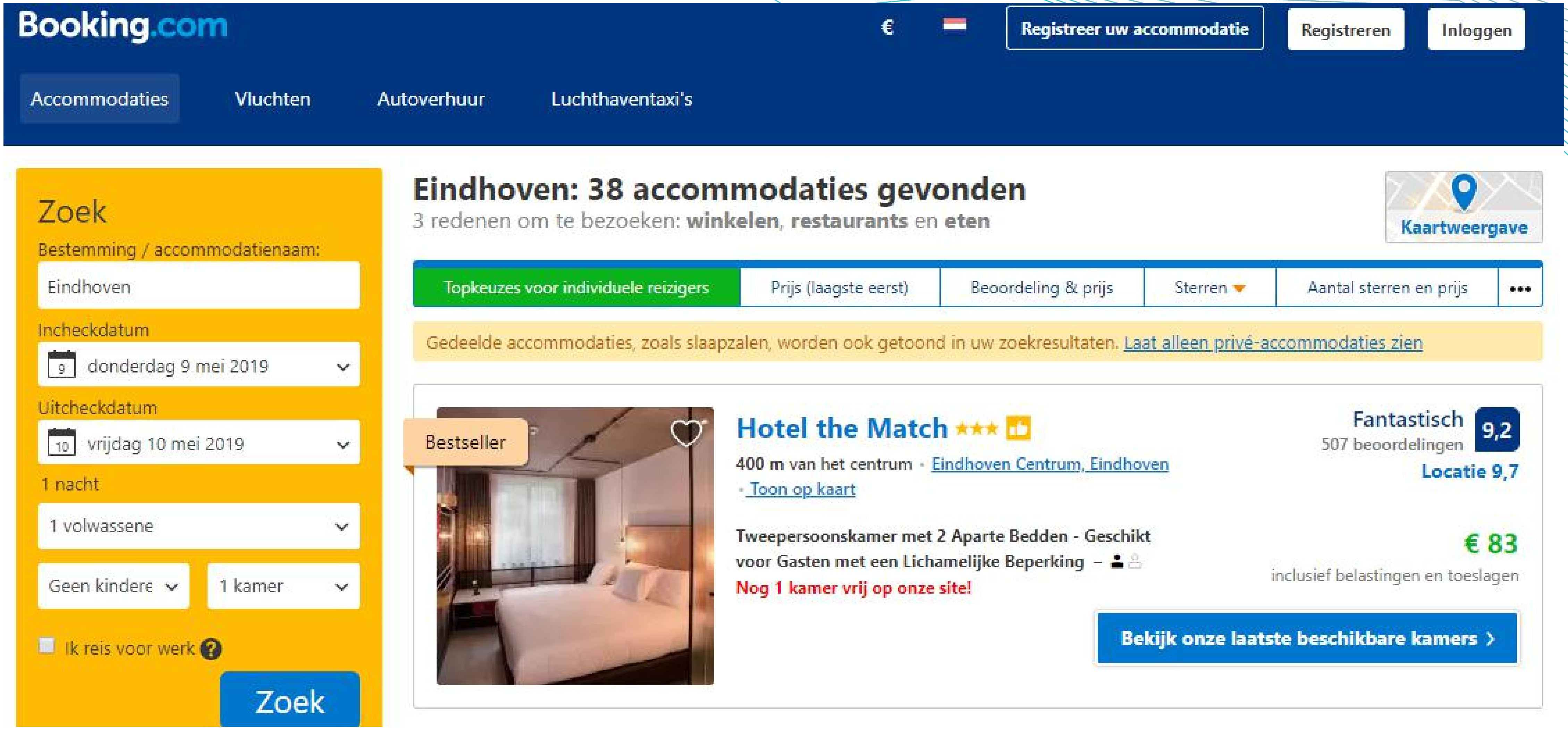
2. Waarom nu?

- Meer data
- Meer computerkracht
- Patronen in grote data sets
- Niet lineaire verbanden
- Kan ge-automatiseerd worden

Machine learning

Machine learning is a subset of artificial intelligence in the field of computer science that often uses statistical techniques to give computers the ability to "learn" with data, without being explicitly programmed.

W Meer op Wikipedia


Booking.com
€
Registreer uw accommodatie
Registreren
Inloggen
Accommodaties
Vluchten
Autoverhuur
Luchthaventaxi's
Zoek
Bestemming / accommodatienaam:
Eindhoven
Incheckdatum
donderdag 9 mei 2019
Uitcheckdatum
vrijdag 10 mei 2019
1 nacht
1 volwassene
Geen kindere
1 kamer
Ik reis voor werk
Zoek
Eindhoven: 38 accommodaties gevonden
3 redenen om te bezoeken: winkelen, restaurants en eten
Kaartweergave
Topkeuzes voor individuele reizigers
Prijs (laagste eerst)
Beoordeling & prijs
Sterren
Aantal sterren en prijs
Gedeelde accommodaties, zoals slaapzalen, worden ook getoond in uw zoekresultaten. Laat alleen privé-accommodaties zien
Bestseller
Hotel the Match
400 m van het centrum · Eindhoven Centrum, Eindhoven
· Toon op kaart
Tweepersoonskamer met 2 Aparte Bedden - Geschikt voor Gasten met een Lichamelijke Beperking –
Nog 1 kamer vrij op onze site!
Fantastisch
507 beoordelingen
9,2
Locatie 9,7
€ 83
inclusief belastingen en toeslagen
Bekijk onze laatste beschikbare kamers

Learning to Match

Themis Mavridis
Booking.com
Amsterdam, Netherlands
themistoklis.mavridis@booking.com

Pablo Estevez
Booking.com
Amsterdam, Netherlands
pablo.estevez@booking.com

Lucas Bernardi
Booking.com
Amsterdam, Netherlands
lucas.bernardi@booking.cc

BSTRACT

ooking.com is a virtual two-sided marketplace where guests and
ccommodation providers are the two distinct stakeholders. They

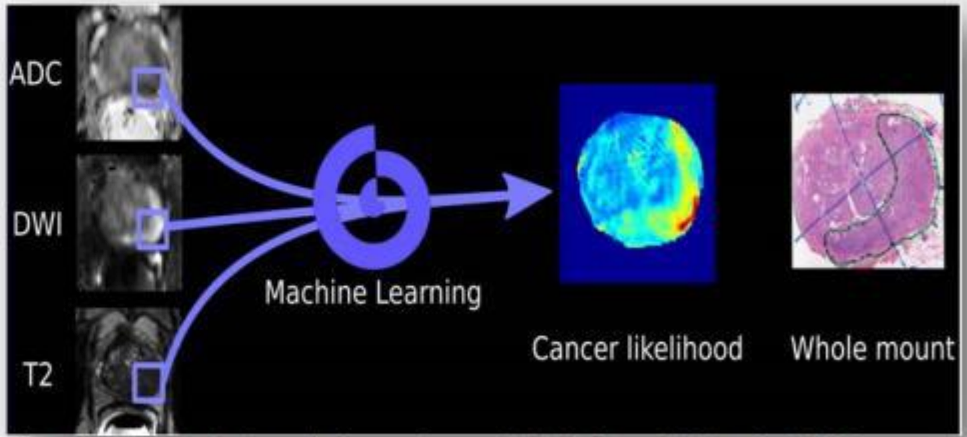
bed and breakfasts, guest houses, etc. The problem of
supply and demand can be approached from several an

a decision maker. Booking.com implements this idea with hundreds of Machine Learned Models, all of them validated through rigorous Randomized Controlled Experiments. We further elaborate on model types, techniques, methodological issues and challenges that we have faced.

Supervised learning

How/Where can it be useful?

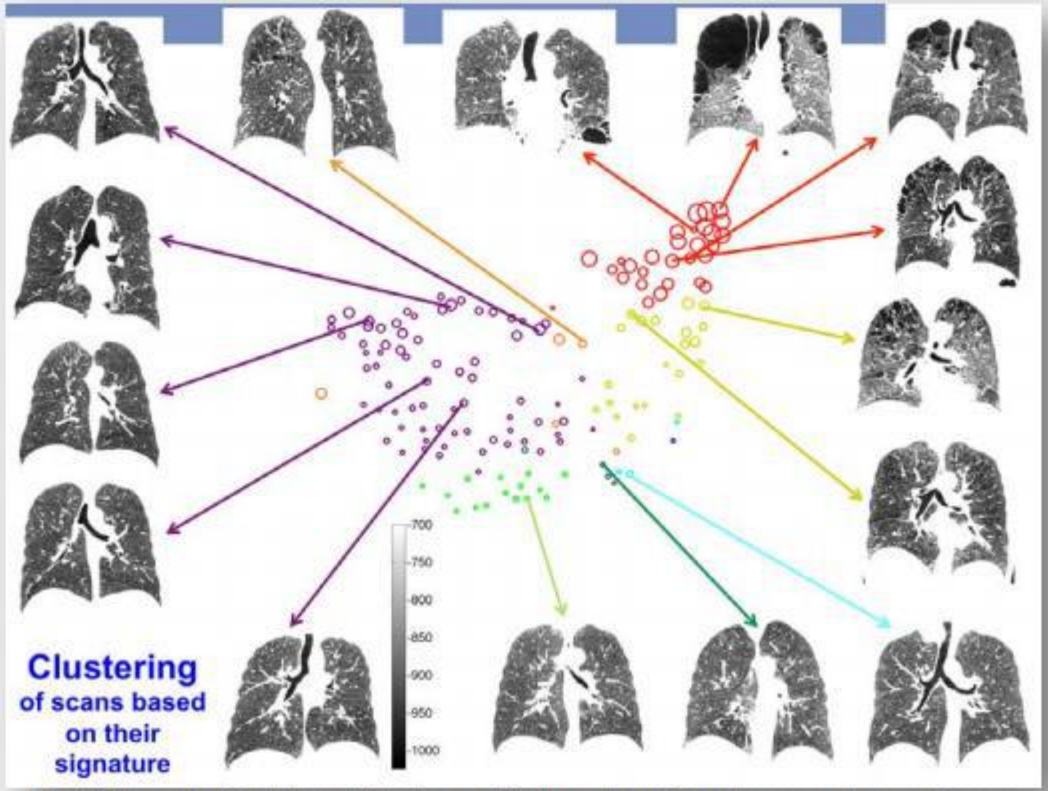
- **Supervised**
  - Learning from [human] expert



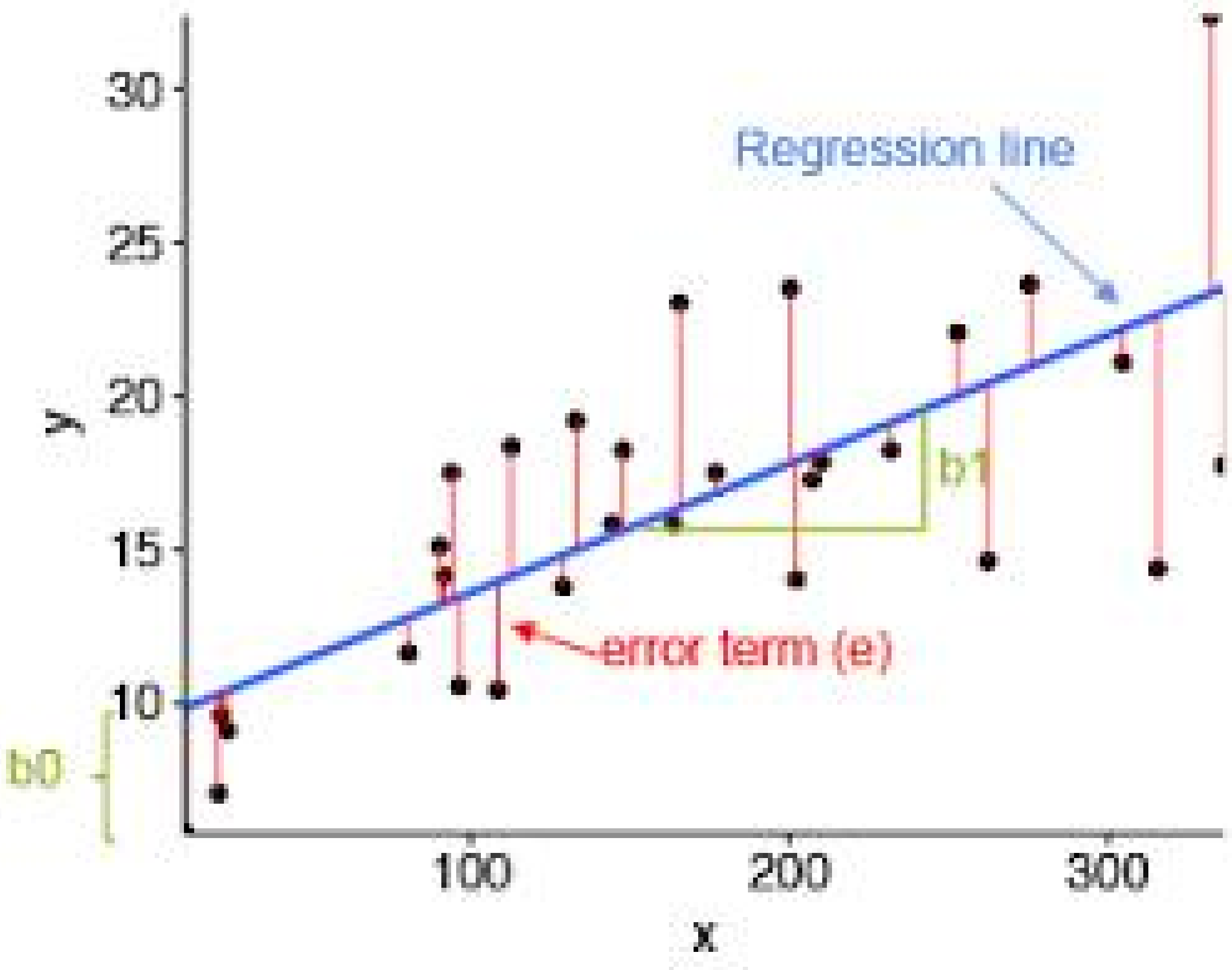
http://www.auntminnie.com/index.aspx?sec=ser&sub=def&pag=dis&ItemID=114700

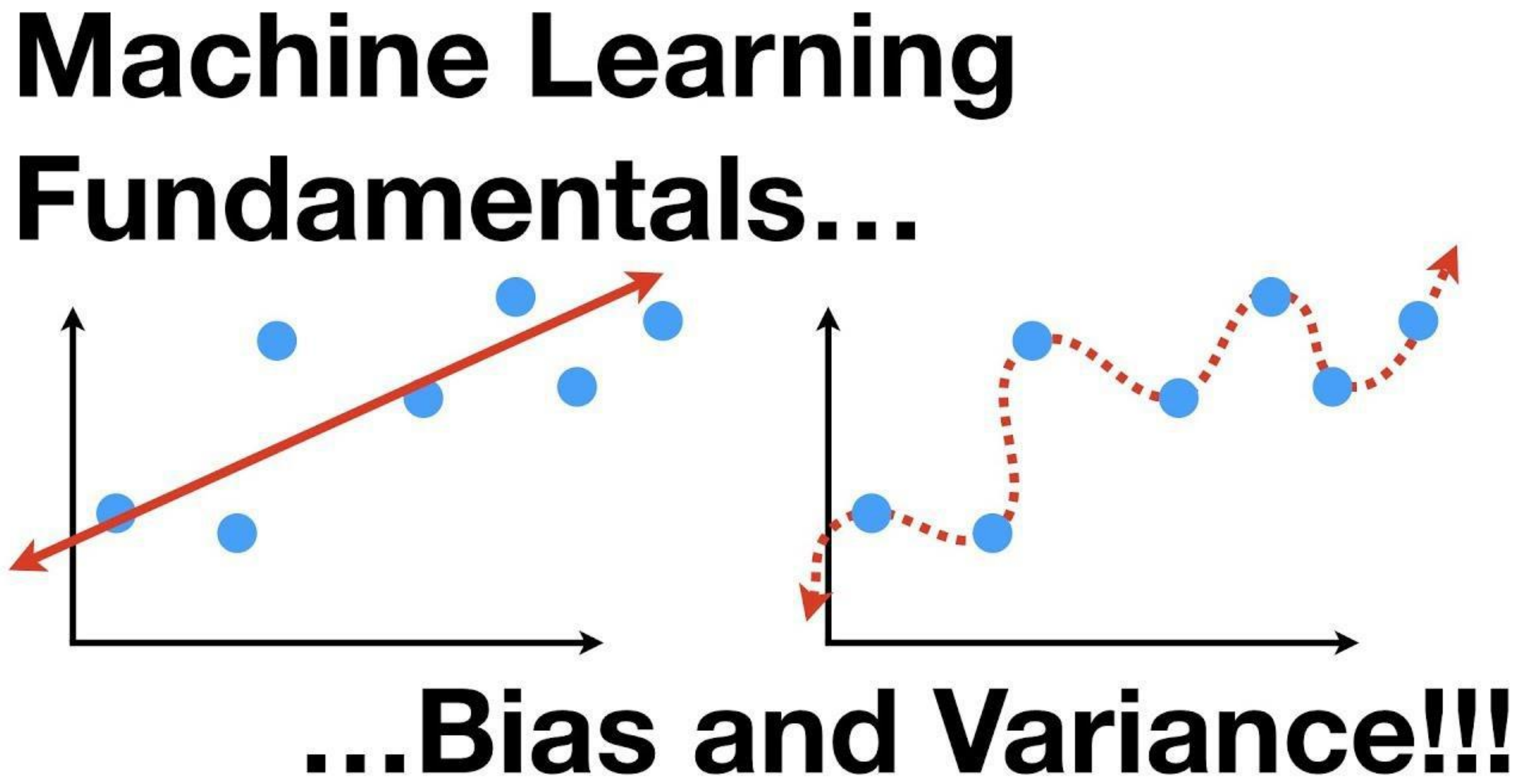
  - PIRADS
  - BIRADS
  - LUNGRADS
  - ...

- **Unsupervised**
  - Discover new knowledge

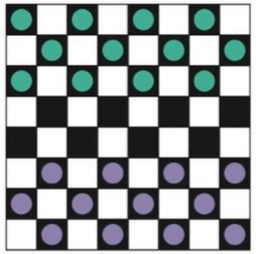


http://hbil.bme.columbia.edu/content/adaptive-quantification-and-subtyping-pulmonary-emphysema-ct-images

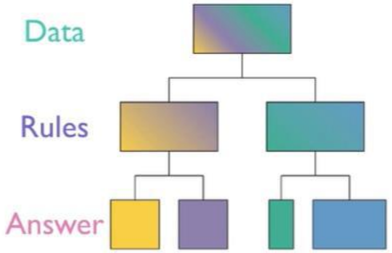

Regression line
error term (e)
b1
b0
y
x
30
25
20
15
10
100
200
300


Machine Learning
Fundamentals...
...Bias and Variance!!!

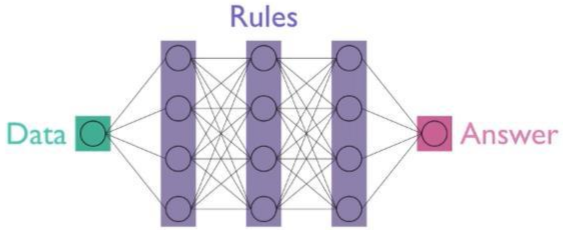
Symbolic AI, Classical Programming

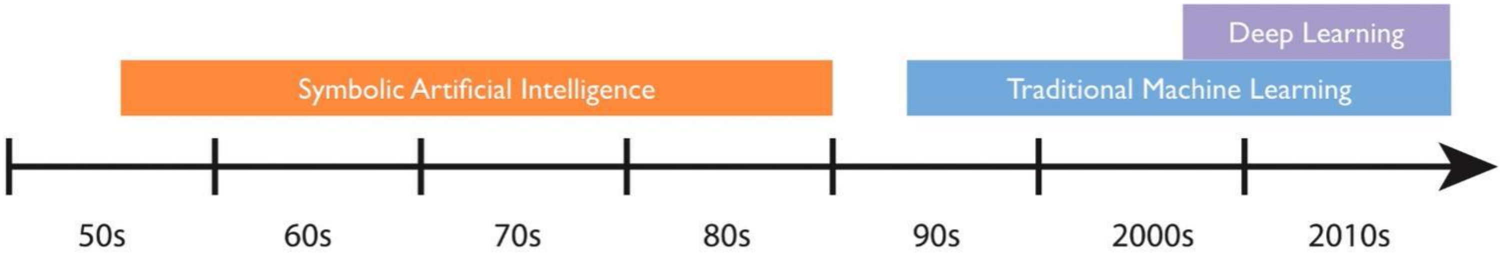


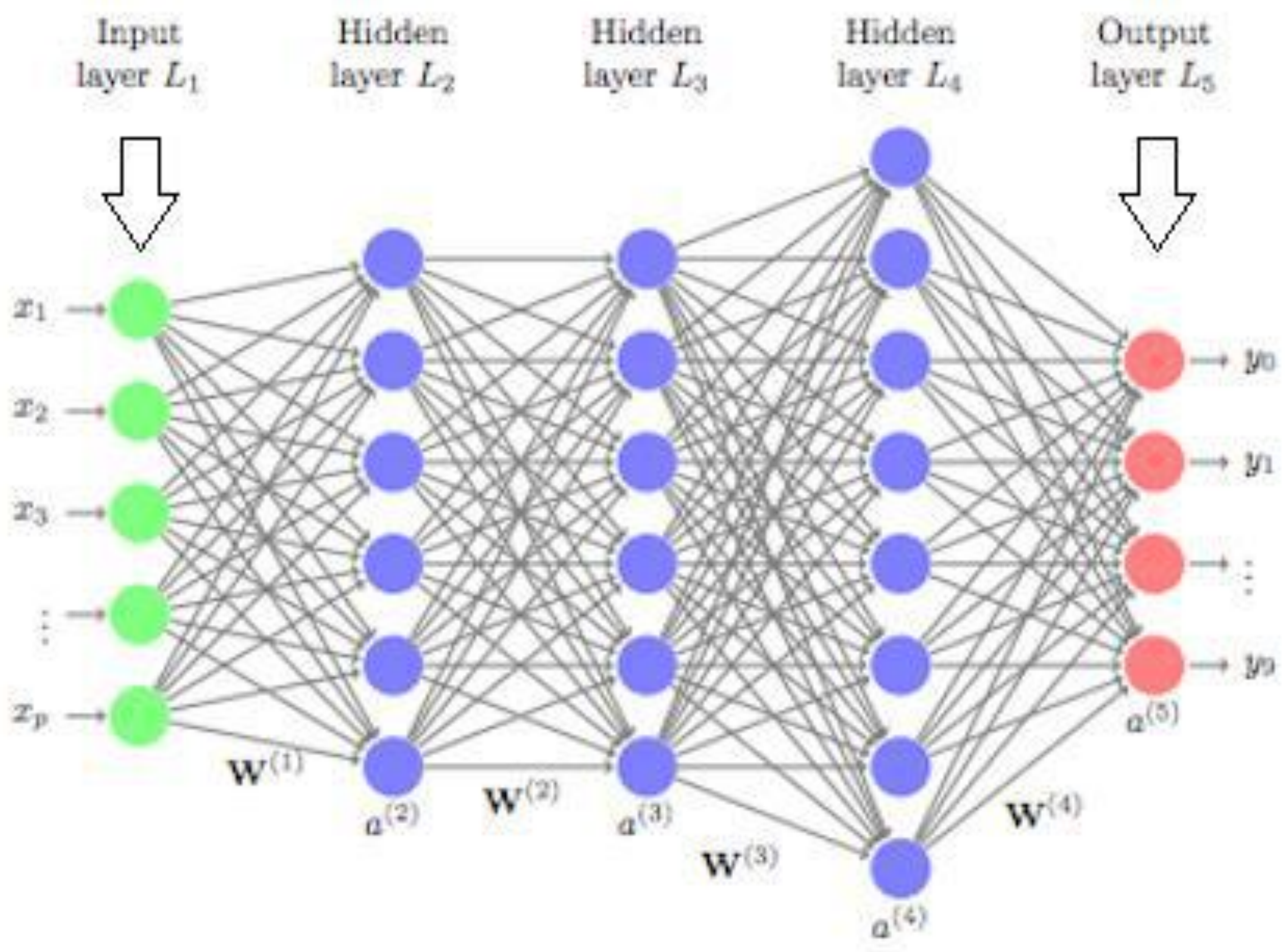
Traditional Machine Learning



Deep Learning

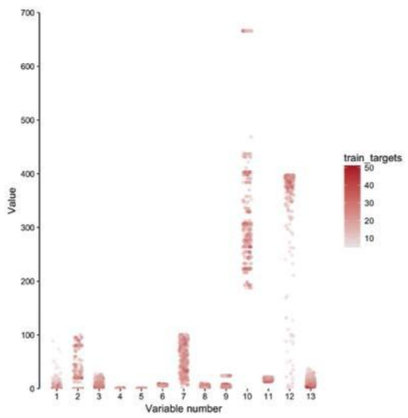





Input layer L_1
Hidden layer L_2
Hidden layer L_3
Hidden layer L_4
Output layer L_5
x_1
x_2
x_3
$\vdots$
x_p
$\mathbf{W}^{(1)}$
$a^{(2)}$
$\mathbf{W}^{(2)}$
$a^{(3)}$
$\mathbf{W}^{(3)}$
$a^{(4)}$
$\mathbf{W}^{(4)}$
$a^{(5)}$
y_0
y_1
$\vdots$
y_9

Supervised Machine Learning

Regression

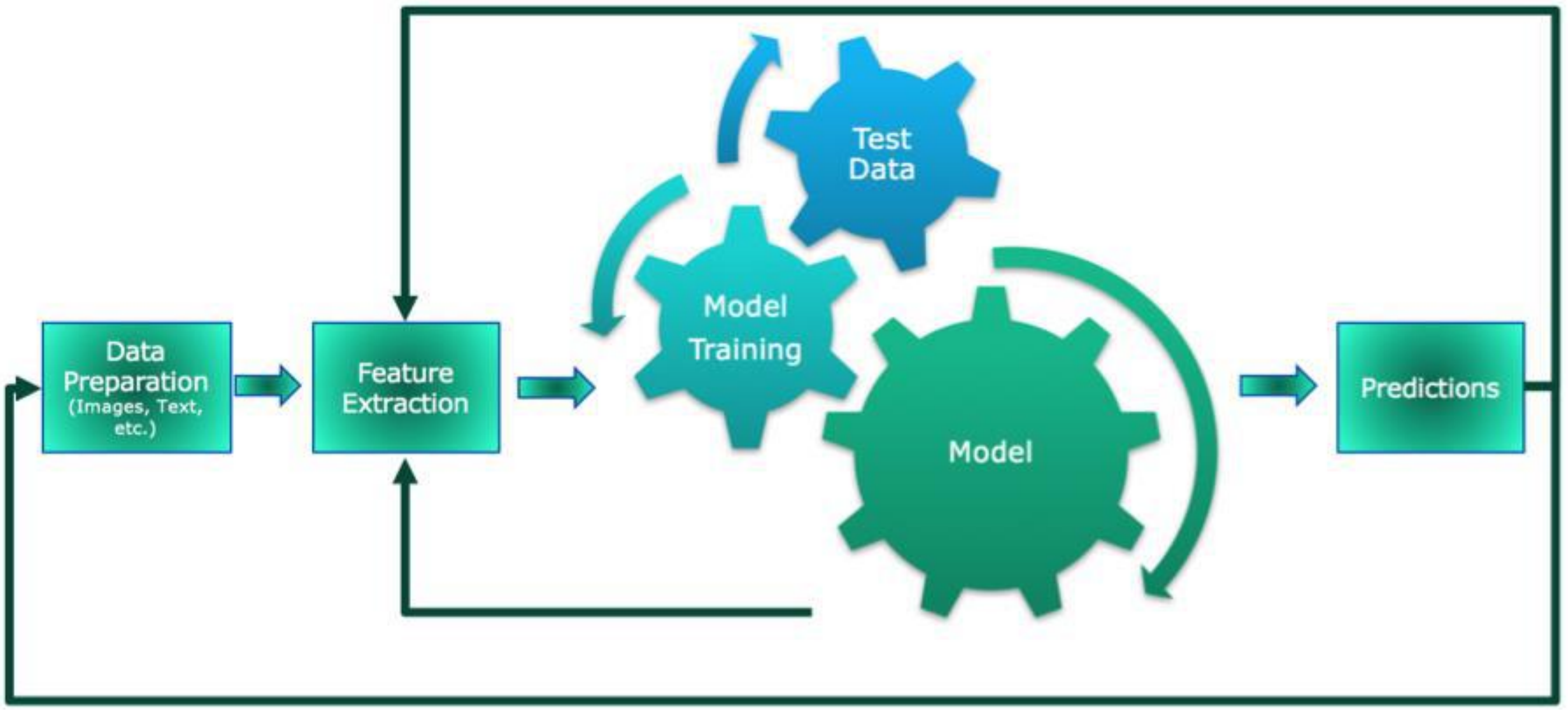


Prediction on a continuous scale

Classification

Prediction on a categorical scale

A Standard Machine Learning Pipeline



Area under the curve

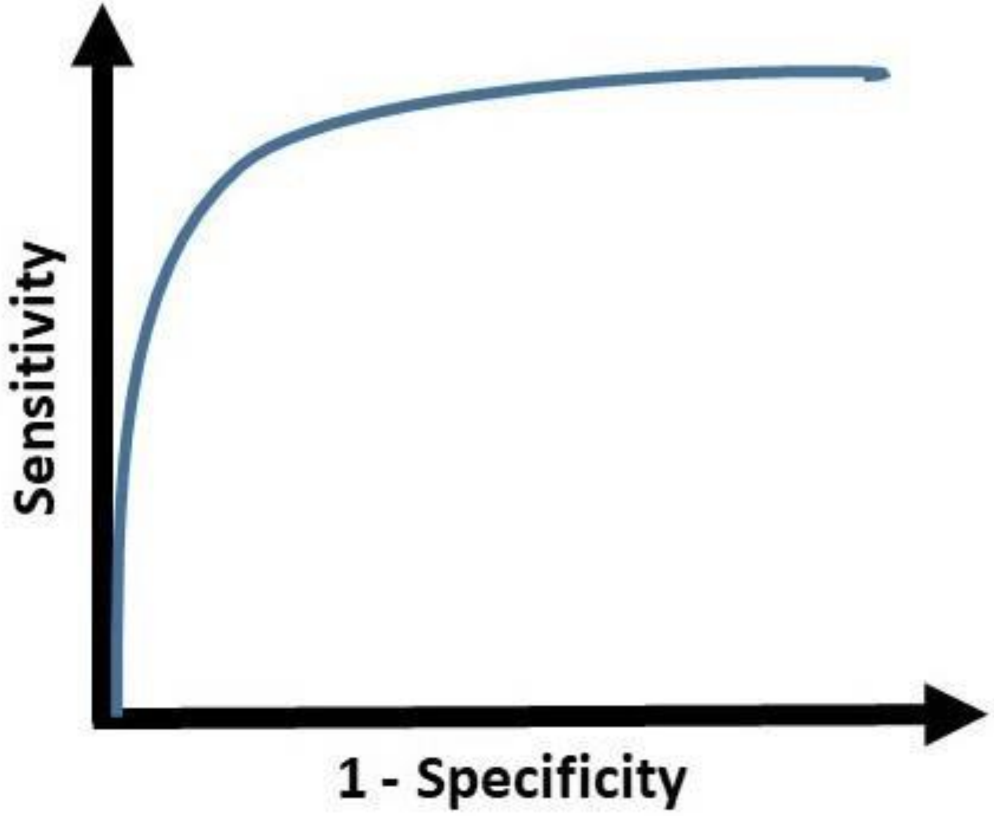
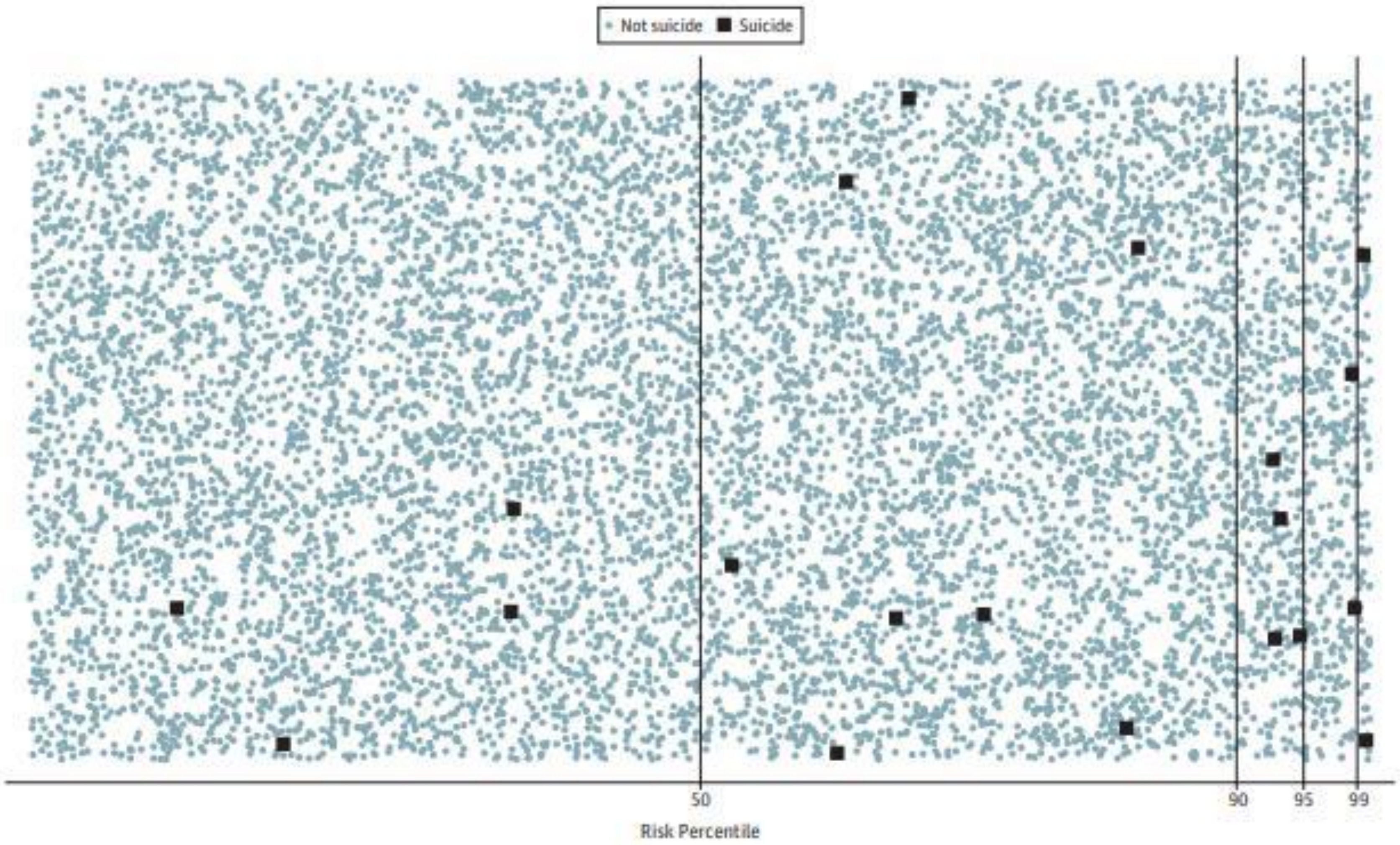


Figure 2. Illustration of Implementing a Suicide Prediction Model



Risk Factors for Suicidal Thoughts and Behaviors: A Meta-Analysis of 50 Years of Research

Joseph C. Franklin and Jessica D. Ribeiro
Vanderbilt University and Harvard University

Kate H. Bentley
Boston University

Xieyining Huang and Katherine M. Musacchio
Vanderbilt University

Bernard P. Chang
Columbia University Medical Center

Kathryn R. Fox
Harvard University

Evan M. Kleiman
Harvard University

Adam C. Jaroszewski
Harvard University

Matthew K. Nock
Harvard University

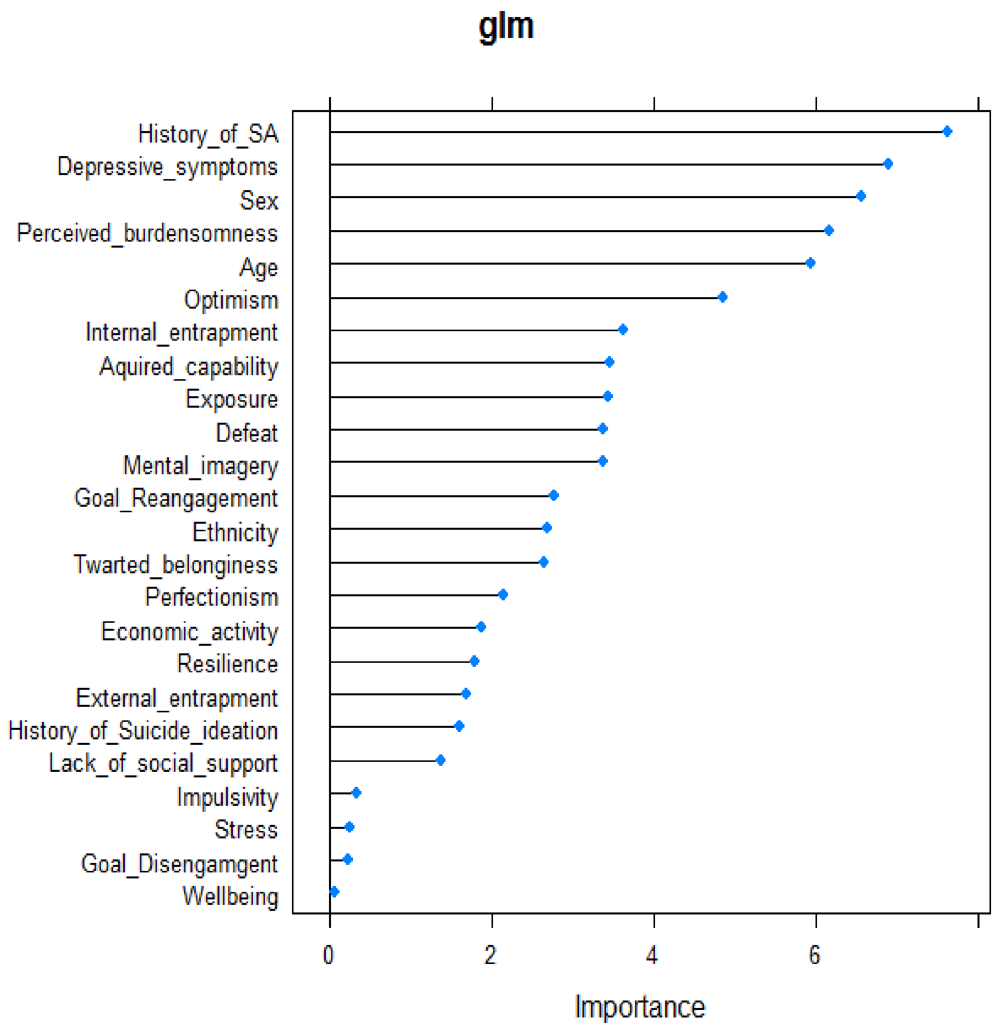
Predicting future suicidal behaviour with different machine learning techniques: a population-based longitudinal study

Kasper van Mens, CWM de Schepper, Ben Wijnen, Saskia J Koldijk, Hugo Schnack, Peter de Looff, Joran Lokkerbol, Karen Wetherall, Seonaid Cleare, Rory C O'Connor, Derek de Beurs.

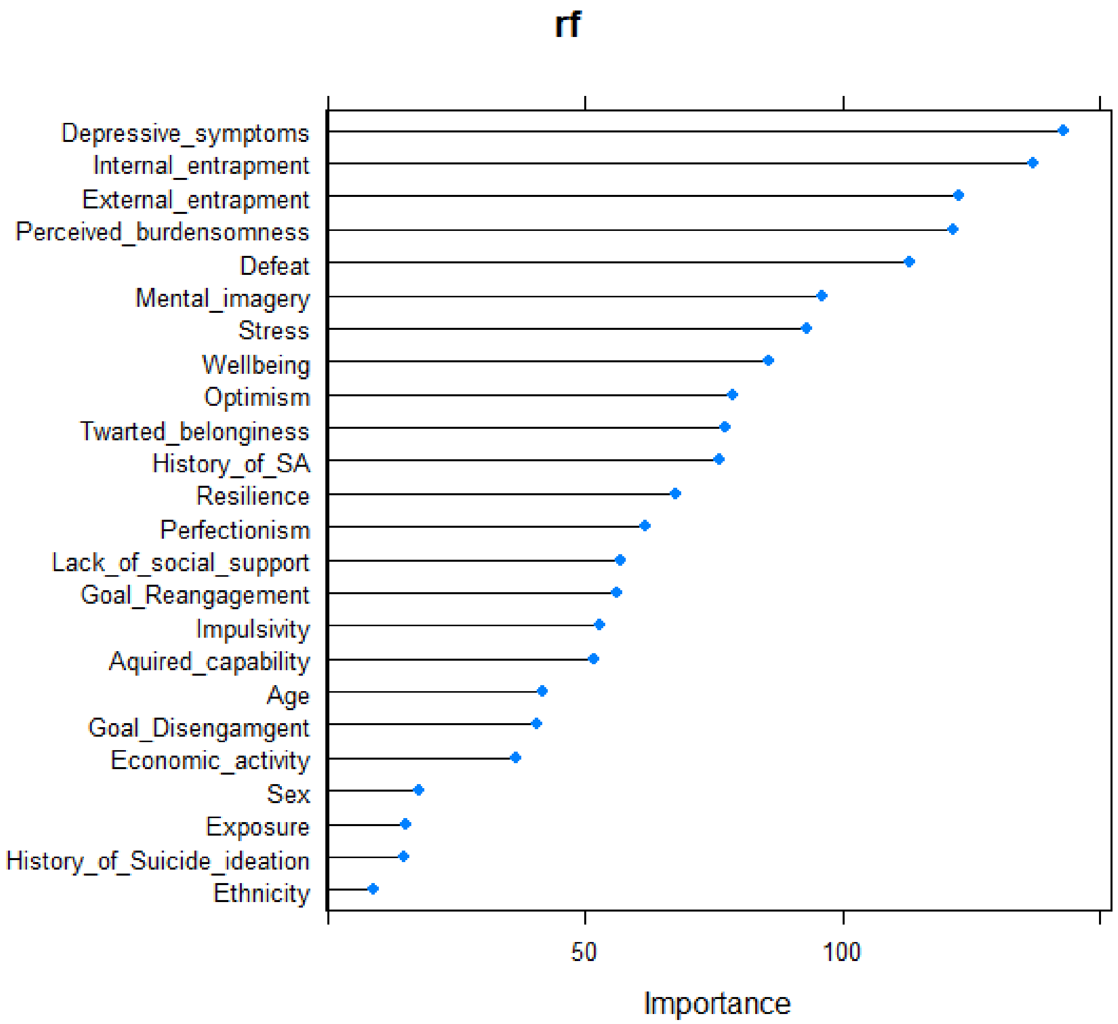
- Population based survey
- 3508 pp at baseline answered a battery of scales on suicide risk factors
- 2426 pp finished one year follow up
- Could we predict suicide ideation (336(14%)) and suicide attempt (50(2%))
- Model 1: 20 sumscores
- Model 2: 20 sumscores + all separate items
- Compared 6 different algoritms

```
###########################################
# PROSPER Hackathon challenge May 2018
###########################################

#start fresh
rm(list = ls())
gc()

library(rdrop2)       #access to dropbox (https://github.com/karthik/rdrop2)
library(DMwR)         #package voor SMOTE algoritme om data te balanceren
library(caret)        #package voor de machine-learning-algoritmes
library(xgboost)      #package voor xgboost-algoritme
library(randomForest) #package voor random-forests-algoritme
library(rpart)        #package voor decision-tree-algoritme
library(ROCR)         #package voor AUC

# models als vector met mogelijke models die allemaal sequenteel gedraaid kunnen worden
# zet 1 model in de lijst om een los model aan te roepen
models <- c("glm",
            "knn",
            "rpart",
            "rf",
            "xgbTree",
  "svmLinear"    #lijkt het soms niet te doen...
  #"svmPoly",    #very slow
 # "svmRadial"
)

# selecteer gewenste uitkomstmaat
# uitkmostmaat 1: Current_Suicide_Ideation
# uitkomstmaat 2: SI_at_follow_up
# uitkomstmaat 3: Suicide_attempt_at_follow_up
# Uitkomstmaat <- "Current_Suicide_Ideation"
Uitkomstmaat <- "SI_at_follow_up"
Uitkomstmaat <- "Suicide_attempt_at_follow_up"


#run scripts:
source('readData.R')          #read from csv file en impute missing values
source('splitting.R')         #split de data in K-fold Cross validation and upsample
source('machine learning.R')  #execute machine learning algorithms (which are specified in models)
```

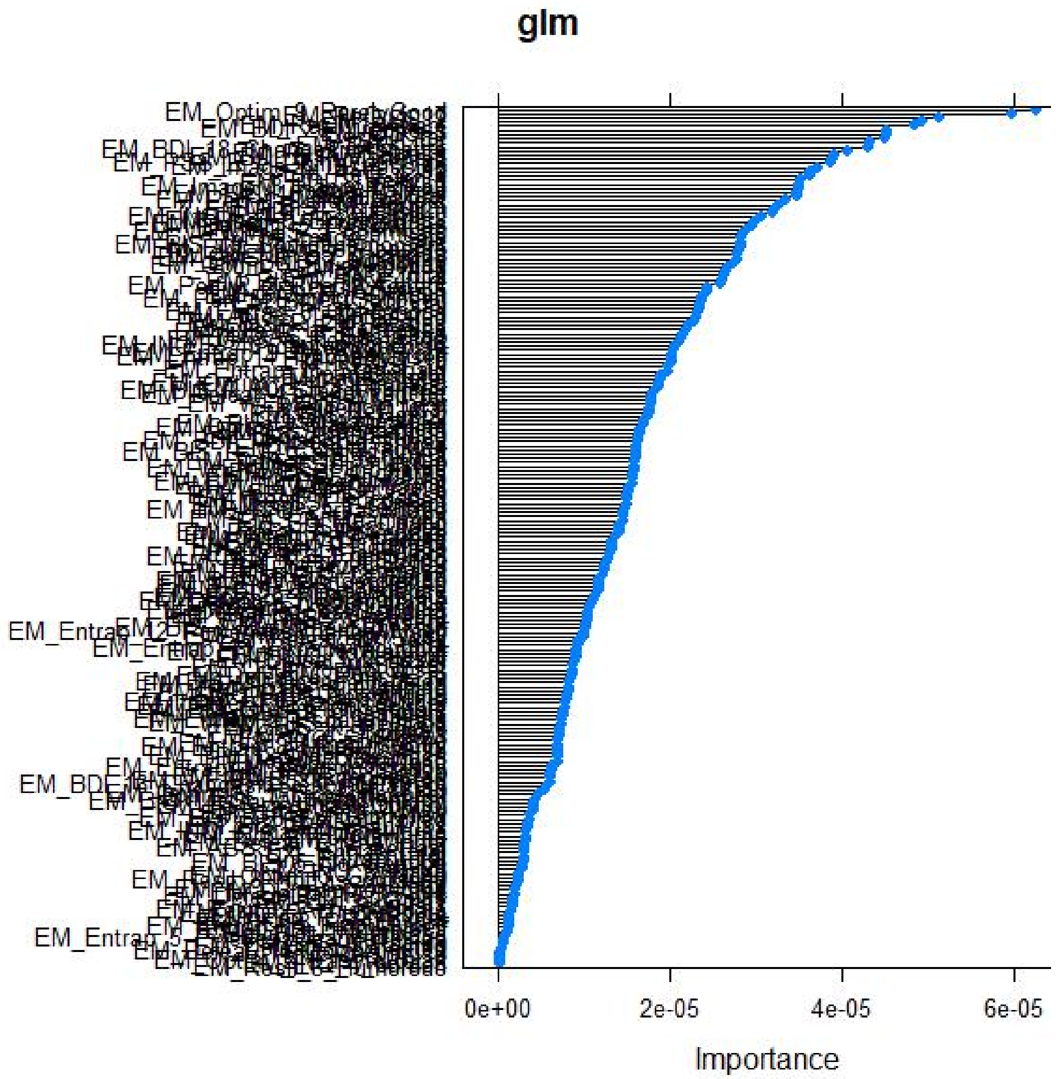
```r
##########################################################
# fit different models
##########################################################
source('evaluate.R')

# cache voor getrainde algoritmes
fits <- list()

##########################################################
### functie

Train_en_evalueer <- function(naam, ...){
  # Maken van extra uitkomst variabel want oa. KNN en XGBoost willen een tekstuele uitkosmtmaat (factor maken was niet genoeg)
  # Maak nieuwe functie als de uitkomstmaat weer numeriek moet zijn
  dataset[dataset[,Uitkomstmaat] == 0, "FactorUitkomst"] <- "Nee"
  dataset[dataset[,Uitkomstmaat] == 1, "FactorUitkomst"] <- "Ja"
  dataset$FactorUitkomst = factor(dataset$FactorUitkomst)

  if (naam %in% models){
    fits[[naam]] <<- caret::train(FactorUitkomst ~ .,
                                  data = dataset[, -which(names(dataset) == Uitkomstmaat)],
                                  method = naam,
                                  trControl=trctrl,
                                  preProcess = c("center", "scale"),
                                  metric = "ROC",
                                  ...
    )

    Print_evaluatie(fits[[naam]], naam)
  }

}

##########################################################
### voor extra algoritmes:  Train_en_evalueer(<naam methode in caret>, <extra parameters>)

# Train_en_evalueer("glm", family = "binomial")
# Train_en_evalueer("knn", tuneLength = 10)
# Train_en_evalueer("rpart")
# Train_en_evalueer("rf")
# Train_en_evalueer("xgbTree")
# Train_en_evalueer("svmLinear", tuneLength = 5)
# Train_en_evalueer("svmPoly", tuneLength = 5)
# Train_en_evalueer("svmRadial", tuneLength = 5)

##########################################################

Train_en_evalueer("glm", family = "binomial")
Train_en_evalueer("knn", tuneLength = 10)
Train_en_evalueer("rpart")
Train_en_evalueer("rf",tuneGrid=expand.grid(mtry=c(5:10)))
Train_en_evalueer("xgbTree",tuneGrid = expand.grid(
```

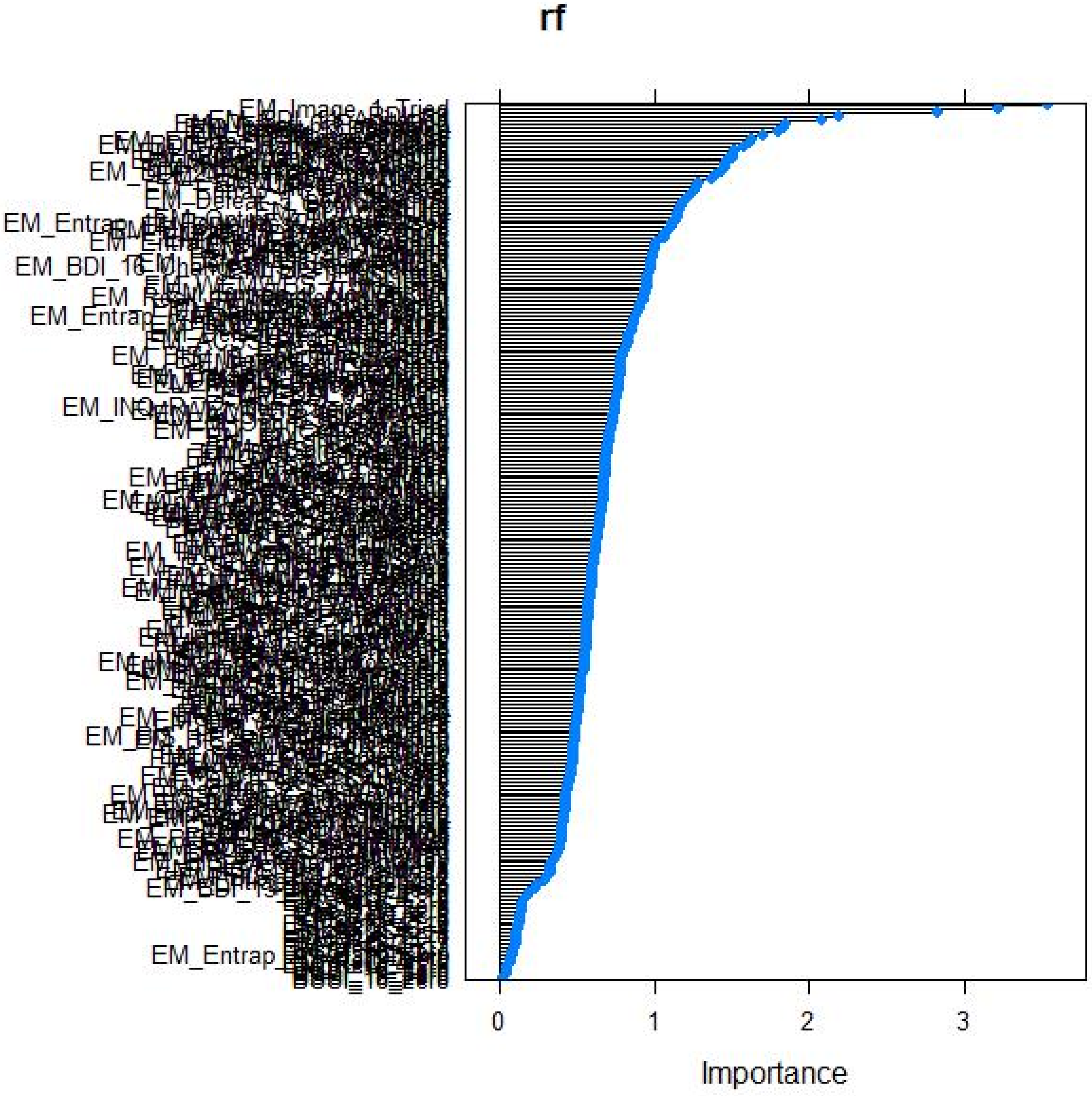


glm
History_of_SA
Depressive_symptoms
Sex
Perceived_burdensomness
Age
Optimism
Internal_entrapment
Aquired_capability
Exposure
Defeat
Mental_imagery
Goal_Reangagement
Ethnicity
Twarted_belonginess
Perfectionism
Economic_activity
Resilience
External_entrapment
History_of_Suicide_ideation
Lack_of_social_support
Impulsivity
Stress
Goal_Disengamgent
Wellbeing
0
2
4
6
Importance



rf
Depressive_symptoms
Internal_entrapment
External_entrapment
Perceived_burdensomness
Defeat
Mental_imagery
Stress
Wellbeing
Optimism
Twarted_belonginess
History_of_SA
Resilience
Perfectionism
Lack_of_social_support
Goal_Reangagement
Impulsivity
Aquired_capability
Age
Goal_Disengamgent
Economic_activity
Sex
Exposure
History_of_Suicide_ideation
Ethnicity
50
100
Importance


glm
0e+00
2e-05
4e-05
6e-05
Importance


rf
EM_Image_1_Tried
0
1
2
3
Importance

Model 1	area under the curve	sensitivity	specificity	positive predictive value
generalized linear model	0.84	0.65	0.85	0.41
K-nearest neighbor	0.80	0.72	0.76	0.32
Regression forest	0.76	0.67	0.81	0.36
Random forest	0.84	0.60	0.78	0.43
gradient boosting	0.81	0.59	0.86	0.41
Support vector machine	0.84	0.68	0.84	0.41

Model 2	area under the curve	sensitivity	specificity	positive predictive value
generalized linear model	0.66	0.58	0.75	0.27
K-nearest neighbor	0.75	0.90	0.37	0.19
regression forest	0.73	0.60	0.85	0.36
Randomforest	0.84	0.55	0.90	0.46
gradient boosting	0.74	0.49	0.87	0.07
Support vector machine	0.75	0.48	0.86	0.07

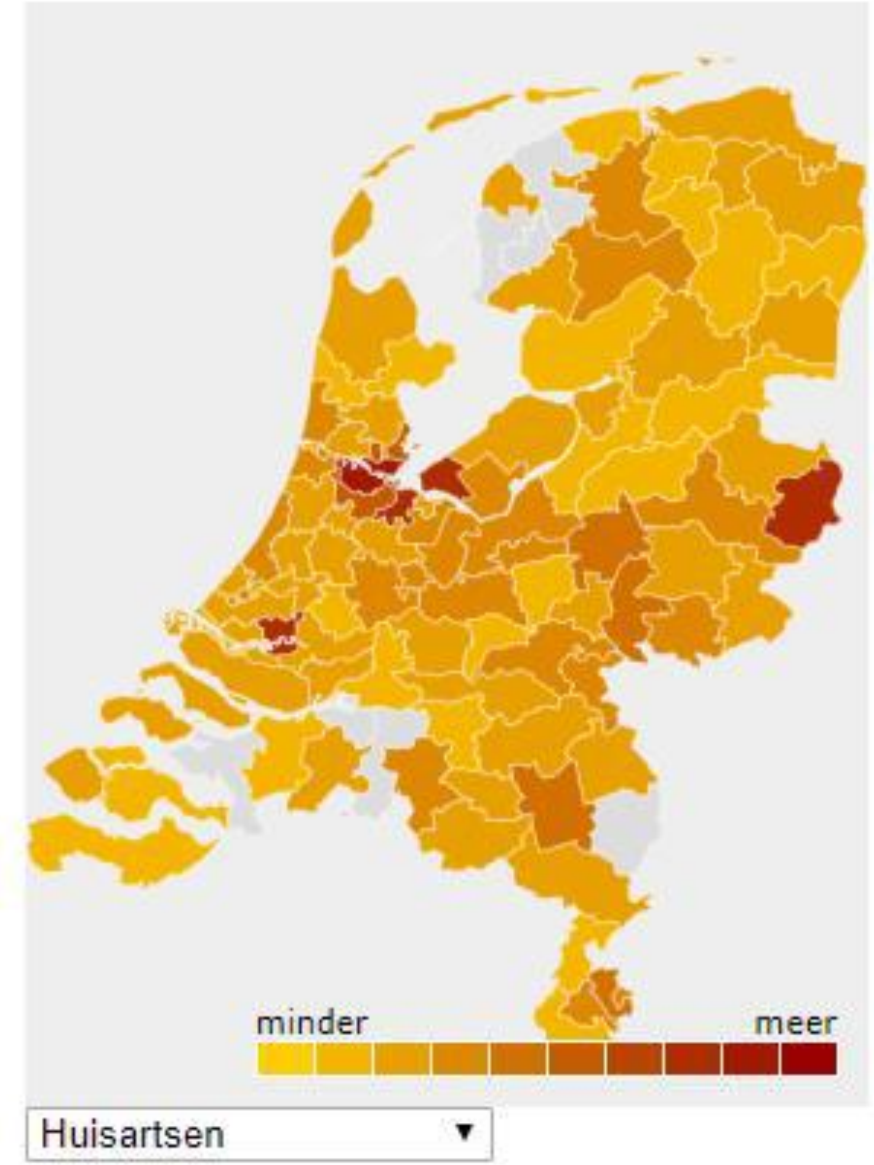
Model 1	area under the curve	sensitivity	specificity	positive predictive value
generalized linear model	0.71	0.54	0.82	0.06
K-nearest neighbor	0.75	0.63	0.79	0.06
regression forest	0.69	0.55	0.80	0.05
Randomforest	0.78	0.51	0.89	0.09
gradient boosting	0.74	0.49	0.87	0.07
Support vector machine	0.75	0.48	0.86	0.07
Model 2	area under the curve	sensitivity	specificity	positive predictive value
generalized linear model	0.57	0.51	0.64	0.03
K-nearest neighbor	0.69	0.70	0.54	0.03
regression forest	0.67	0.42	0.81	0.04
Randomforest	0.78	0.42	0.92	0.10
gradient boosting	0.73	0.42	0.90	0.08
Support vector machine	0.64	0.34	0.88	0.06

Its difficult to be beat logistical regression

- Initial step taken by relatively simple models at the start already explains so much variance
- Our data simply did not capture the necessary constructs accurately enough to model their interaction.
- All algorithms are based on the assumption that there are no errors in the classification, or in the assessment of psychological constructs.
- It is intrinsically difficult to predict future human behaviour

Nivel zorgregistraties eerste lijn

- 420 Huisartsenpraktijken
- 1.7 miljoen patienten
- Diagnoses van patienten (ICPC)
- Longitudinal data (vanaf 2011)




KPMG
Monitor generalistische basis GGZ
Verslagperiode: 2011-2016
In opdracht van het Ministerie van VWS
Juni 2018

Applying machine learning on health record data from general practitioners to predict suicidality

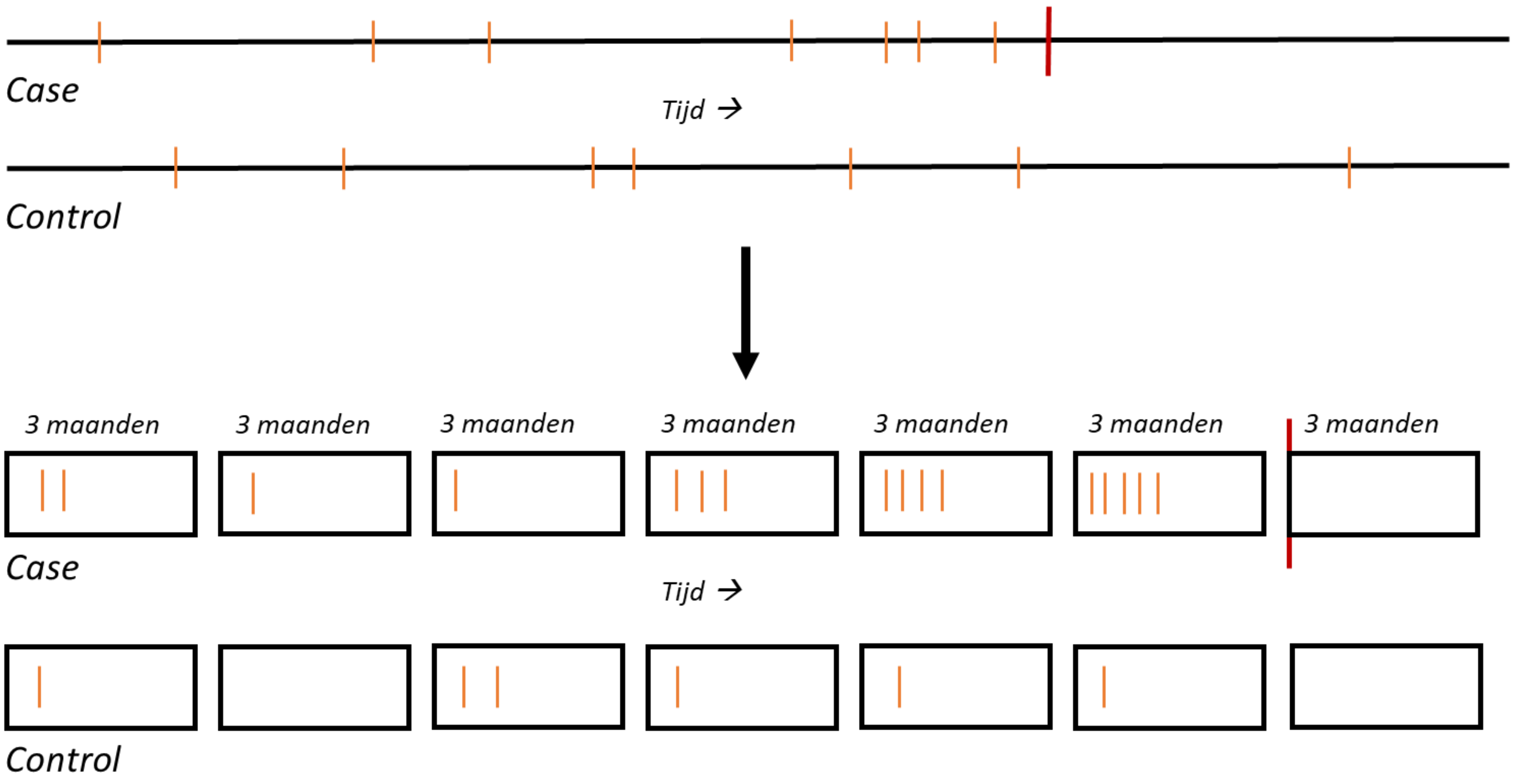
Kasper Mens, Elke Elzinga, Mark Nielen, Joran Lokkerbol, Rune Poortvliet, Gé Donker, Marianne Heins, Joke Korevaar, Michel Dückers, Claire Aussems, Marco Helbich, Bea Tiemens, Renske Gilissen, Aartjan Beekman, Derek de Beurs

Keywords: suicide, general practice, electronic health records, machine learning

Design

- Cases: all patients with a registration of P77, and no registration in the 2 years before (n = 534)
- Controls: patients with at least one consultation for psychological problems and no P77 (n = 35.000)

Dataset prepareren



Topic of last registration (chapter)	Cases
Depression (P)	53 (10%)
Chronic Alcohol abuse (P)	16 (3%)
Diabetes (Other)	14 (3%)
Affective Psychosis (P)	13 (2%)
Personality Disorder (P)	13 (2%)
No disease (Other)	11 (2%)
Essential Hypertension (Other)	11 (2%)
Crisis / stress reaction (P)	10 (2%)
Other psychological symptoms (P)	10 (2%)
Anxiety (P)	10 (2%)

Rank	Variable
1	Relative healthcare uptake (all registrations) 1 month before compared to baseline
2	Number of P-registrations 1 month before
3	Age
4	Relative healthcare uptake MUPS-registrations 1 month before compared to baseline
5	Number of MUPS-registrations 1 month before
6	Relative healthcare uptake P-registrations 1 month before compared to baseline
7	Number of depression registrations 1 month before
8	Relative healthcare uptake (all registrations) 3 months before compared to

	Actual Case	Actual control
Predicted Case	63	1298
Predicted Control	98	52368

	Random Forest
Area under the curve (95% CI)	0.82 (0.78 – 0.86)
Sensitivity	0.39 (0.32 – 0.47)
Specificity	0.98 (0.97 – 0.98)
PPV	0.05 (0.04 – 0.06)
Balanced Accuracy	0.68

ML better predicts suicidal behavior

Predicting suicide attempts in adolescents with longitudinal clinical data and machine learning

Colin G. Walsh,[1] **Jessica D. Ribeiro,**[2] **and Joseph C. Franklin**[2]
[1]Vanderbilt University Medical Center, Nashville, TN; [2]Florida State University, Tallahassee, FL, USA

Lezen van paper

- Lees de abstract, en bespreek het samen met je buurman

Key points

- This study developed machine learning algorithms to detect risk for suicide attempts among adolescents using only routinely collected clinical electronic health record data.
- By combining risk factors including comorbidities, medication usage, clinical encounter histories, socio-economic status, and demographics, machine learning produced accurate prediction across multiple cohort comparisons and time points.
- Applying machine learning to large and widely available clinical data may be a promising avenue toward scalable risk detection in the context of well-designed clinical decision support.

Background: Adolescents have high rates of nonfatal suicide attempts, but clinically practical risk prediction remains a challenge. Screening can be time consuming to implement at scale, if it is done at all. Computational algorithms may predict suicide risk using only routinely collected clinical data. We used a machine learning approach validated on longitudinal clinical data in adults to address this challenge in adolescents. **Methods:** This is a retrospective, longitudinal cohort study. Data were collected from the Vanderbilt Synthetic Derivative from January 1998 to December 2015 and included 974 adolescents with nonfatal suicide attempts and multiple control comparisons: 496 adolescents with other self-injury (OSI), 7,059 adolescents with depressive symptoms, and 25,081 adolescent general hospital controls. Candidate predictors included diagnostic, demographic, medication, and socioeconomic factors. Outcome was determined by multiexpert review of electronic health records. Random forests were validated with optimism adjustment at multiple time points (from 1 week to 2 years). Recalibration was done via isotonic regression. Evaluation metrics included discrimination (AUC, sensitivity/specificity, precision/recall) and calibration (calibration plots, slope/intercept, Brier score). **Results:** Computational models performed well and did not require face-to-face screening. Performance improved as suicide attempts became more imminent. Discrimination was good in comparison with OSI controls (AUC = 0.83 [0.82–0.84] at 720 days; AUC = 0.85 [0.84–0.87] at 7 days) and depressed controls (AUC = 0.87 [95% CI 0.85–0.90] at 720 days; 0.90 [0.85–0.94] at 7 days) and best in comparison with general hospital controls (AUC 0.94 [0.92–0.96] at 720 days; 0.97 [0.95–0.98] at 7 days). Random forests significantly outperformed logistic regression in every comparison. Recalibration improved performance as much as ninefold – clinical recommendations with poorly calibrated predictions can lead to decision errors. **Conclusions:** Machine learning on longitudinal clinical data may provide a scalable approach to broaden screening for risk of nonfatal suicide attempts in adolescents. **Keywords:** Suicide; attempted; adolescent; machine learning; decision support techniques; electronic health records.

Learn about ML!

RStudio®

python™

DataCamp

All Machine Learning Courses

Introduction to Machine Learning

Learn to train and assess models performing common machine learning tasks such as classification and clustering.

Continue Course

Machine Learning Toolbox

This course teaches the big ideas in machine learning like how to build and evaluate predictive models.

Continue Course

Unsupervised Learning in R

This course provides an intro to clustering and dimensionality reduction in R from a machine learning perspective.

4 hours Play preview

HANK ROARK
Senior Data Scientist, Boeing

Supervised Learning with scikit-learn

Learn how to build and tune predictive models and evaluate how well they will perform on unseen data.

4 hours Play preview

HUGO BOWNE-ANDERSON
Data Scientist at DataCamp

Deep Learning in Python

Learn the fundamentals of neural networks and how to build deep learning models using Keras 2.0.

Continue Course

Unsupervised Learning in Python

Learn how to cluster, transform, visualize, and extract insights from unlabeled datasets using scikit-learn and scipy.

4 hours Play preview

BENJAMIN WILSON
Director of Research at lateral.io

Classification models

- Categorical (i.e. qualitative) target variable
- Example: will a loan default?
- Still a form of supervised learning
- Use a train/test split to evaluate performance
- Use the `Sonar` dataset

DataCamp

Course Outline

Exercise

Plot an ROC curve

As you saw in the video, an ROC curve is a really useful shortcut for summarizing the performance of a classifier over all possible thresholds. This saves you a lot of tedious work computing class predictions for many different thresholds and examining the confusion matrix for each.

My favorite package for computing ROC curves is `caTools`, which contains a function called `colAUC()`. This function is very user-friendly and can actually calculate ROC curves for multiple predictors at once. In this case, you only need to calculate the ROC curve for one predictor, e.g.:

```
colAUC(predicted_probabilities, actual, plotROC = TRUE)
```

The function will return a score called AUC (more on that later) and the `plotROC = TRUE` argument will return the plot of the ROC curve for visual inspection.

Instructions 0XP

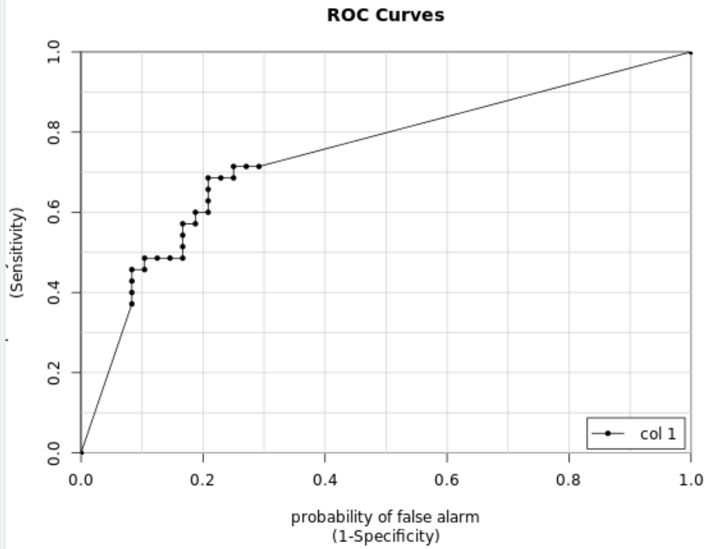
`model`, `test`, and `train` from the last exercise using the sonar data are loaded in your workspace.

script.R solution.R

```
# Predict on test: p
p <- predict(model, test, type = "response")

# Make ROC curve
colAUC(p, test[["Class"]], plotROC = TRUE)
```

Run Code Run Solution

Plots

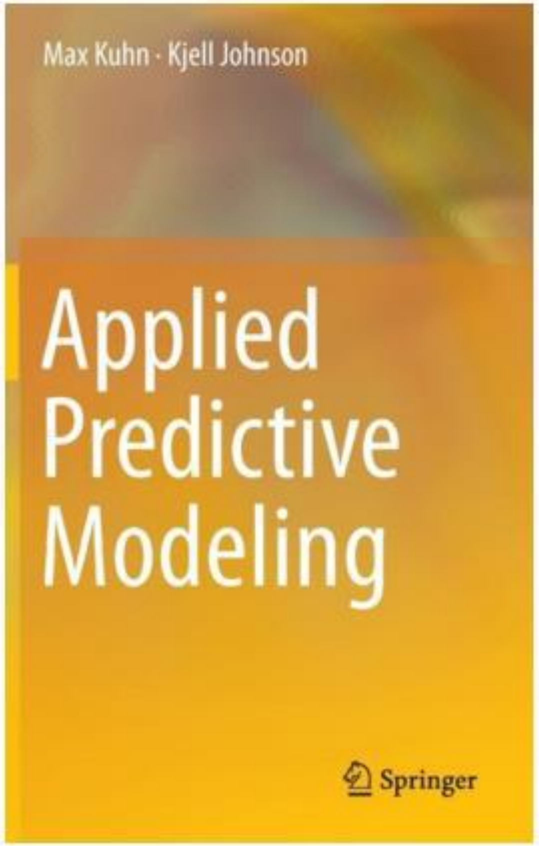


Previous Plot 1/1 Next Plot

R Console Slides

https://topepo.github.io/caret/index.html

Kuhn | Taal: Engels | Schrijf een review |



The `caret` Package

Max Kuhn

2019-03-27

1 Introduction

The `caret` package (short for **C**lassification **A**nd **RE**gression **T**raining) is a set of functions that attempt to streamline the process for creating predictive models. The package contains tools for:

- data splitting
- pre-processing
- feature selection
- model tuning using resampling
- variable importance estimation

https://projectflutrend.github.io/

- Summary
  - TL;DR: Code only version
  - Results at a glance: 'Nowcasting' Influenza in Germany
- Scope
- Get data
- Pre-processing
- Model building
- Results
- Discussion

- Work in Progress -

Using Wikipedia and Google data to estimate near real-time influenza incidence in Germany: A Tutorial in R

Paul Schneider, *Maastricht University, Netherlands Institute of Health Service Research*
John Paget, *Netherlands Institute of Health Service Research*
Peter Spreeuwenberg, *Netherlands Institute of Health Service Research*
David Barnett, *Maastricht University*
Christel van Gool, *Maastricht University*

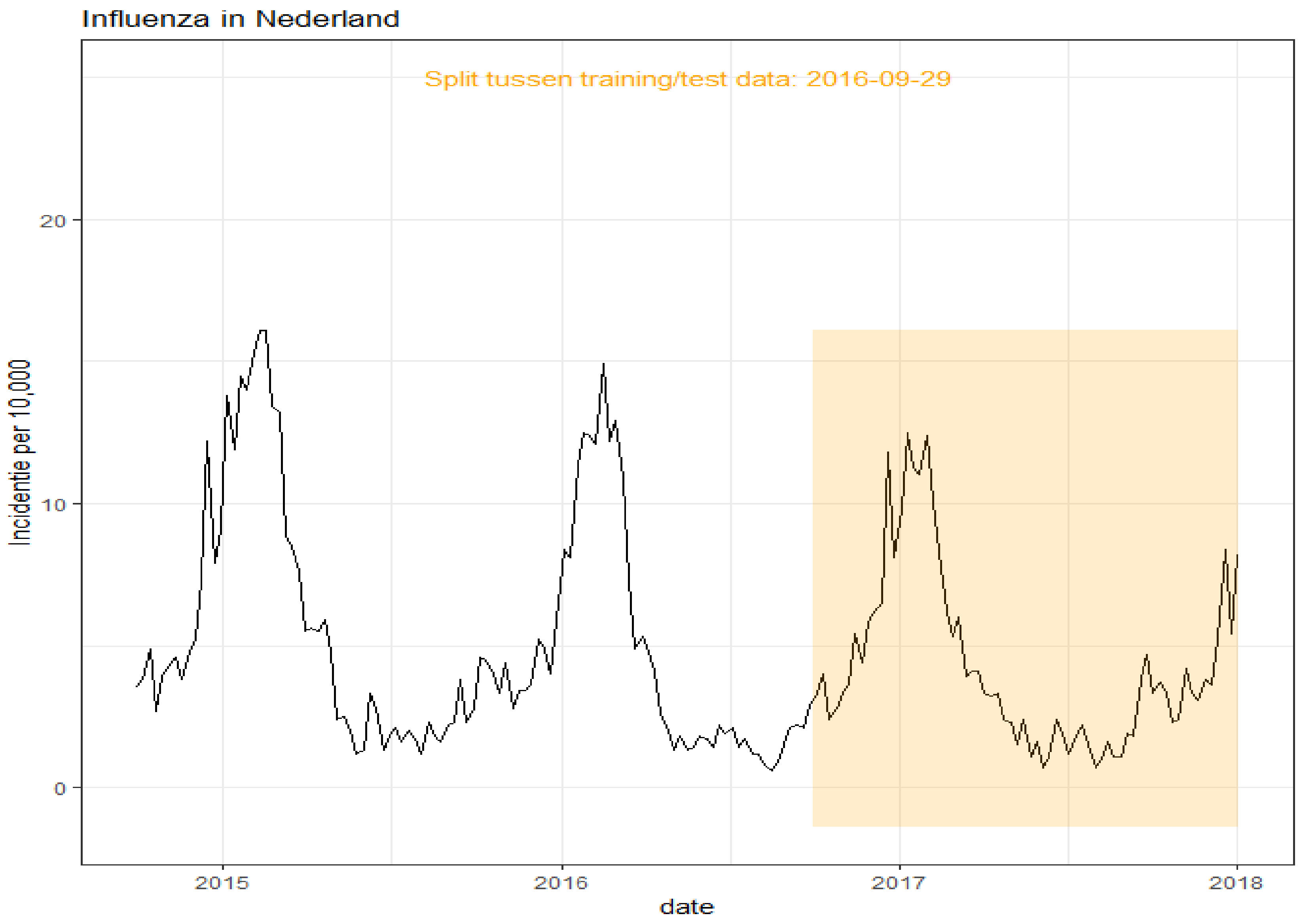
Contact: schneider.paulpeter@gmail.com

Summary

Traditional surveillance systems are costly and involve considerable delay between disease onset and reporting. Previous studies have demonstrated that it is possible to predict the incidence of influenza from relevant Google search queries and Wikipedia page view statistics. Here, we present our approach on how to build a near real-time ('Nowcast') prediction model for monitoring the incidence of influenza in Germany using the statistical software R. Source code and data are fully available and can be re-used, adjusted and transferred to other settings.

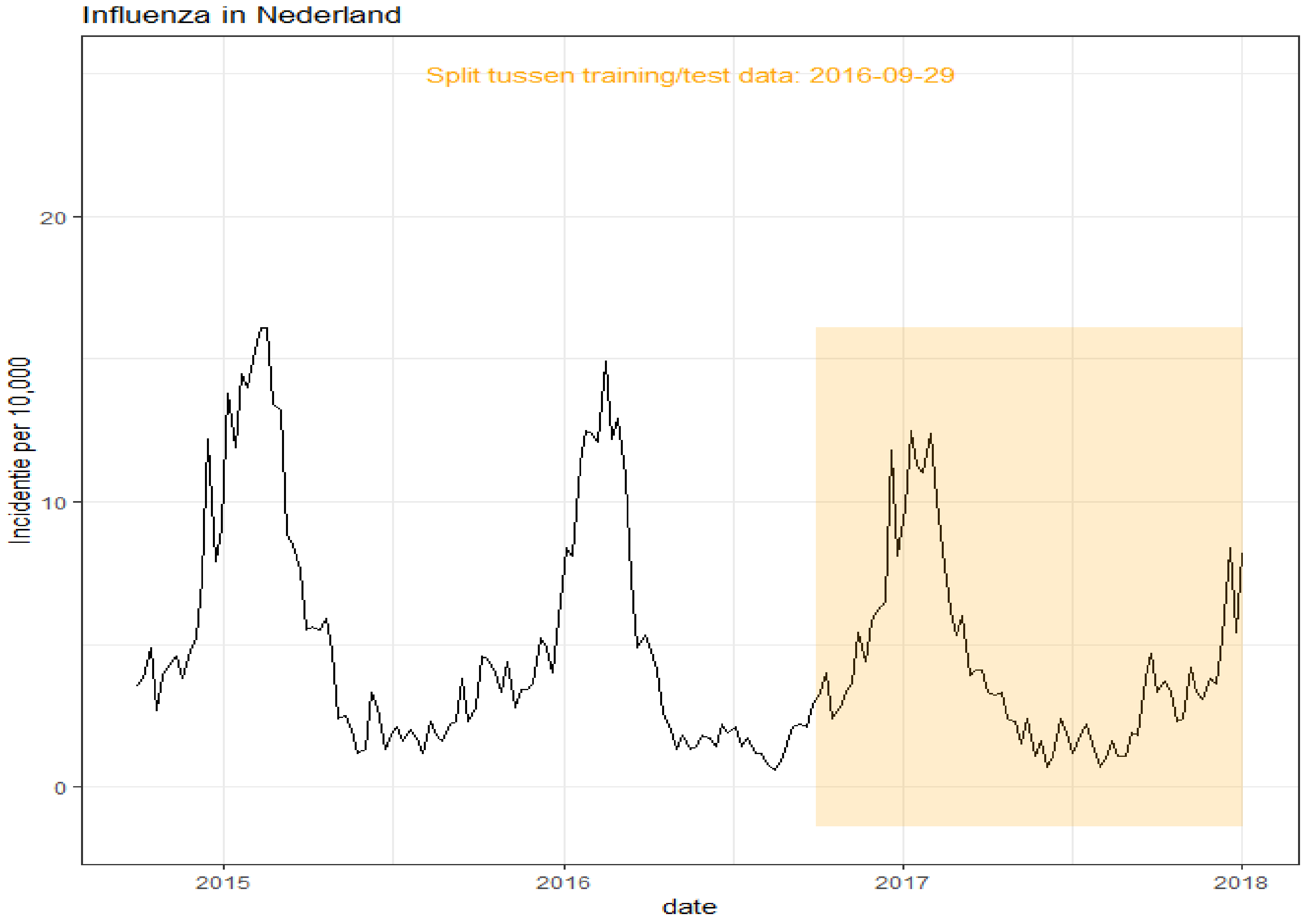
Also see our research paper on this topic: In preparation
And have a look at our Github page

TL;DR: Code only version

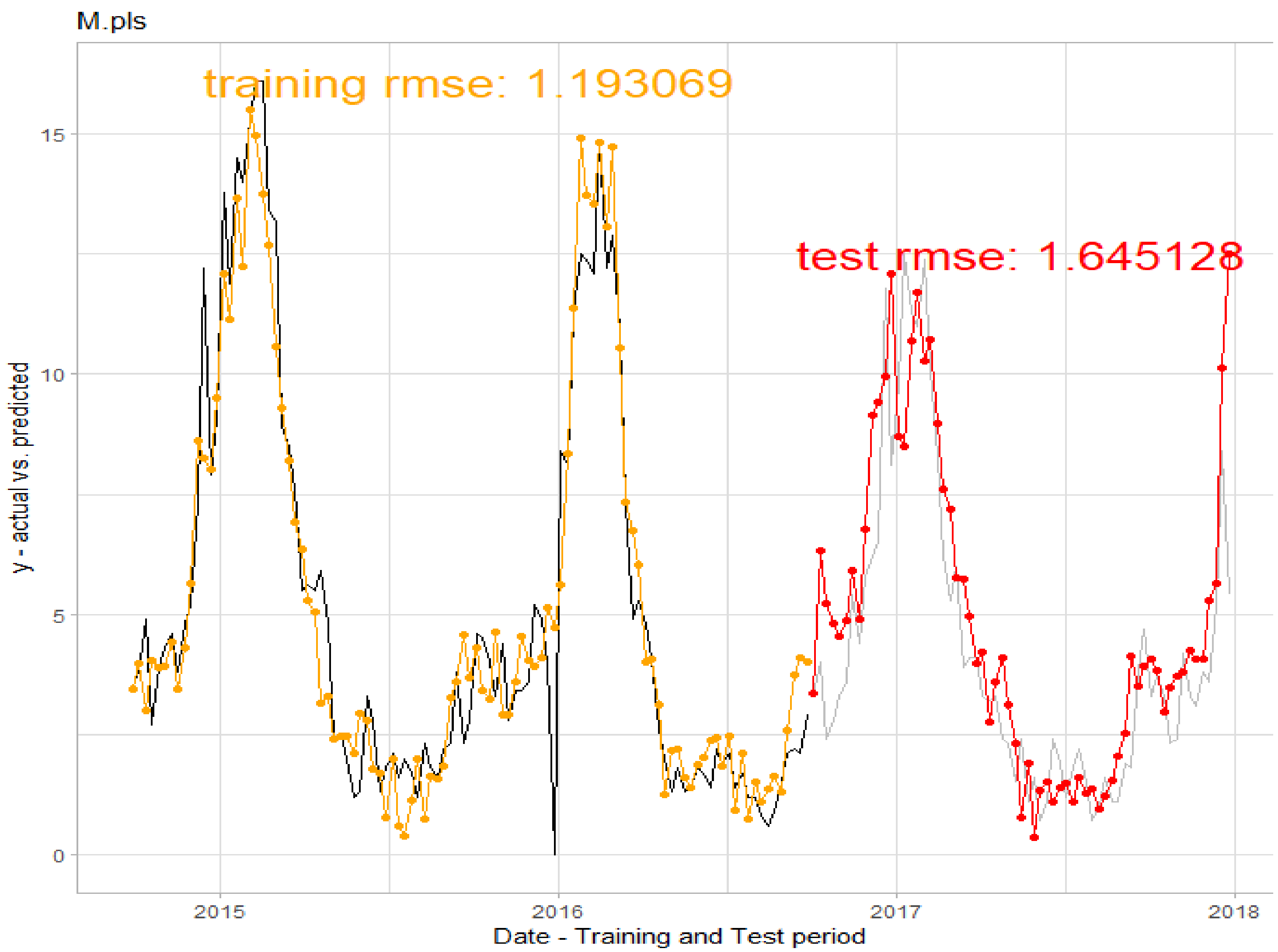

Influenza in Nederland
Split tussen training/test data: 2016-09-29
Incidentie per 10,000
20
10
0
2015
2016
2017
2018
date

	Model: R-function
1	Partial least squares: pls
2	Ridge regression: enet
3	Lasso regression: glmnet
4	Multivariate adaptive regression splines: earth
5	Support vector machine: svmradial
6	Single trees: rpart
7	Single trees: ctree
8	Boosted trees: gbm
9	Bagged trees: treebag
10	Random forest: rf
11	Cubist: cubist
12	Neural Network: AvNNet

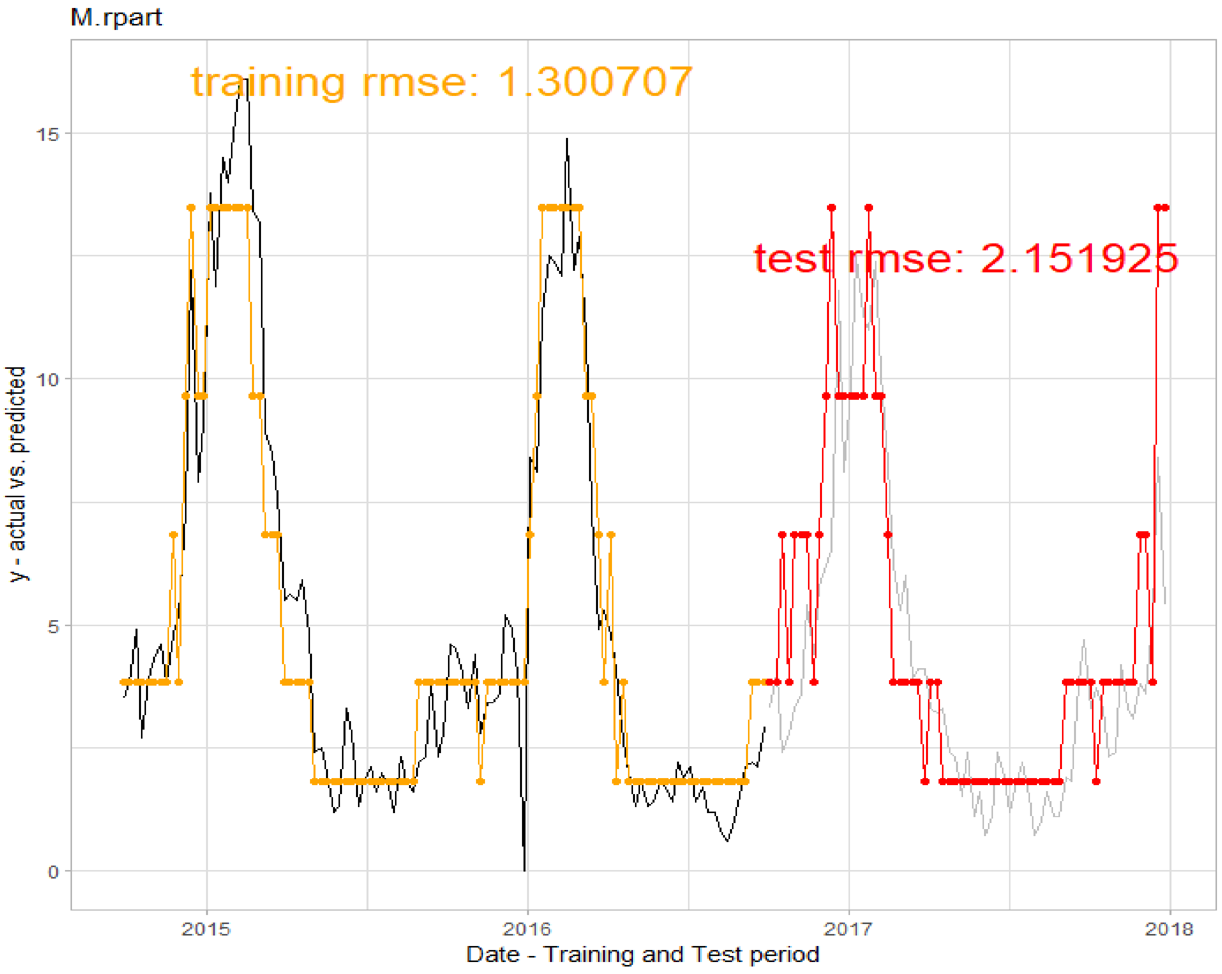
```r
# lasso regression (glmnet)
  lassoGrid <- expand.grid(.alpha = c(.2, .4, .6, .8),.lambda = seq(.05, 1.5, length = 50))
  # Model
  M.lasso <- train(y= y.train ,
                   x = df.train,
                   method = "glmnet",
                  family = "gaussian", # tried poisson, worse!
                  tuneGrid = lassoGrid,
                  trControl = controlObject)

# multivariate adaptive regression splines (earth)
  marsGrid <- expand.grid(.degree = 1, .nprune = 2:15)
  # Model
  M.mars=train(y= y.train ,
          x = df.train,
          method = "earth",
          tuneGrid = marsGrid,
          trControl = controlObject)
```



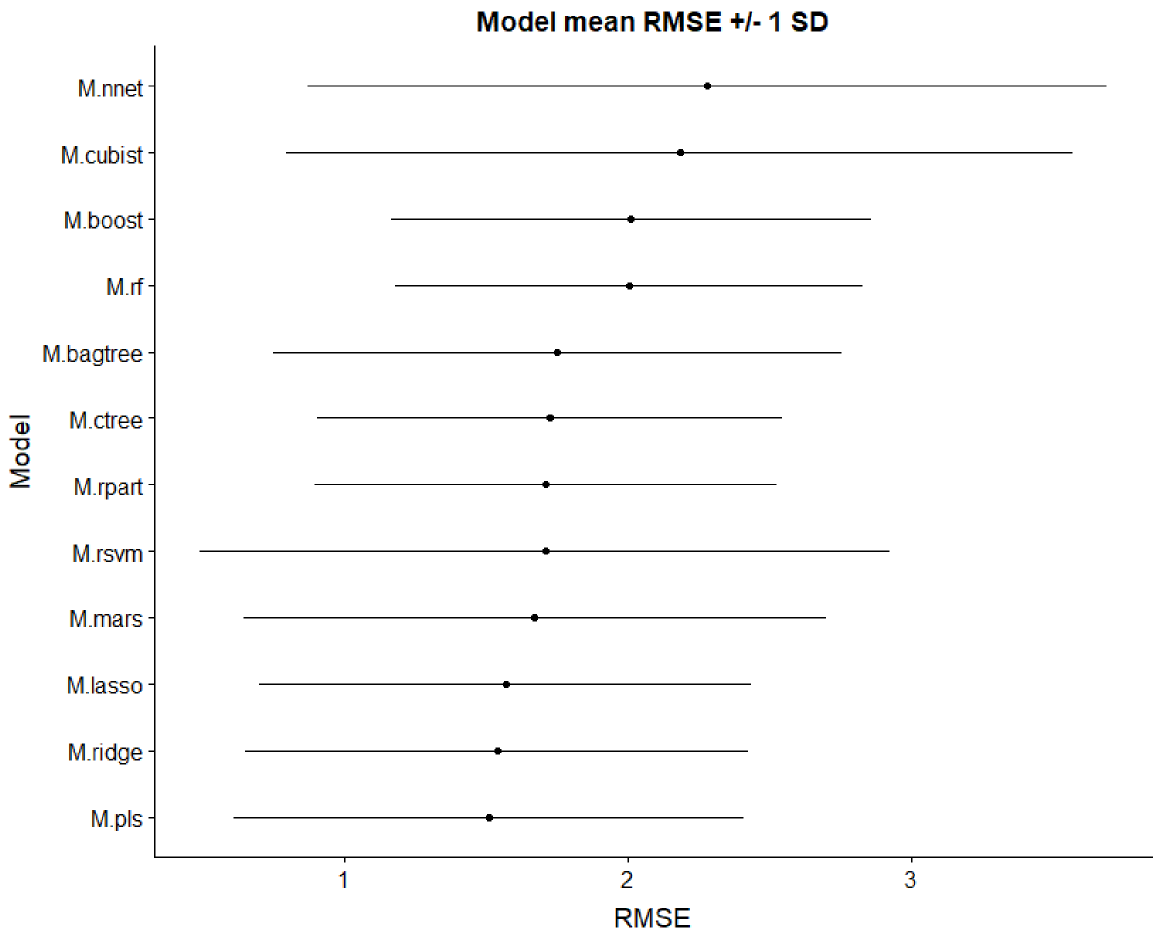
Influenza in Nederland
Split tussen training/test data: 2016-09-29
20
10
0
Incidentie per 10,000
2015
2016
2017
2018
date


M.pls
training rmse: 1.193069
test rmse: 1.645128
15
10
5
0
y - actual vs. predicted
2015
2016
2017
2018
Date - Training and Test period

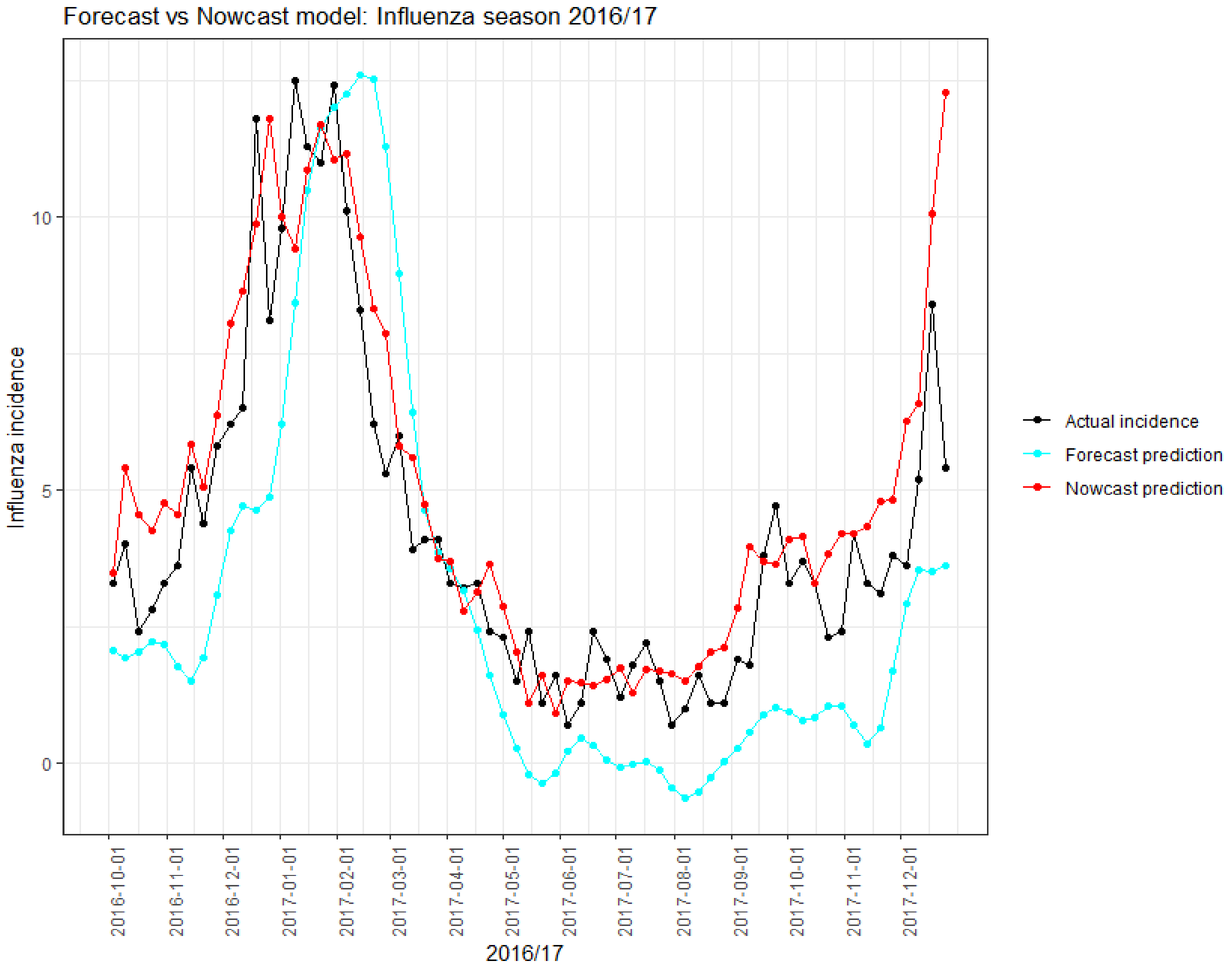
Single regression trees model



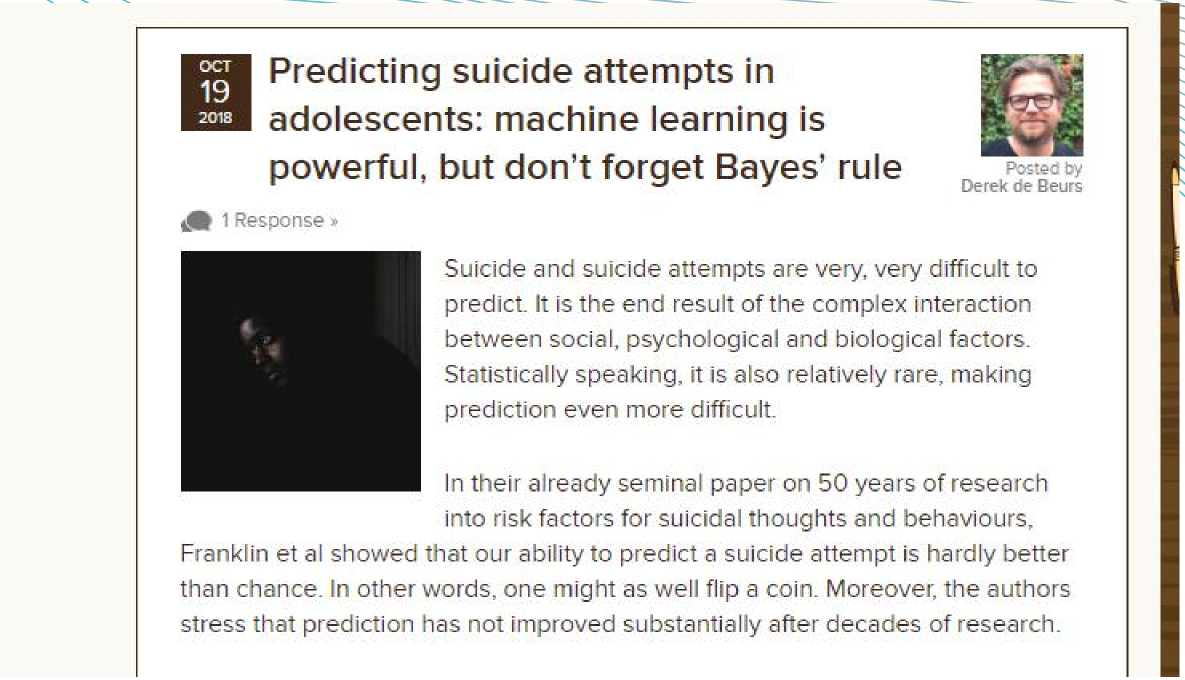
Vergelijking alle modellen



4.9 Vergelijken met nul model



Low base rate


OCT 19 2018

Predicting suicide attempts in adolescents: machine learning is powerful, but don't forget Bayes' rule

Posted by Derek de Beurs

1 Response »

Suicide and suicide attempts are very, very difficult to predict. It is the end result of the complex interaction between social, psychological and biological factors. Statistically speaking, it is also relatively rare, making prediction even more difficult.

In their already seminal paper on 50 years of research into risk factors for suicidal thoughts and behaviours, Franklin et al showed that our ability to predict a suicide attempt is hardly better than chance. In other words, one might as well flip a coin. Moreover, the authors stress that prediction has not improved substantially after decades of research.

Sensitiviteit 0.39
Specificiteit 0.96
Pos Pred Value 0.07
Neg Pred Value 0.99

Algoritm will detect 140 cases, of which 10 will be true positives

JAMA Psychiatry | Review

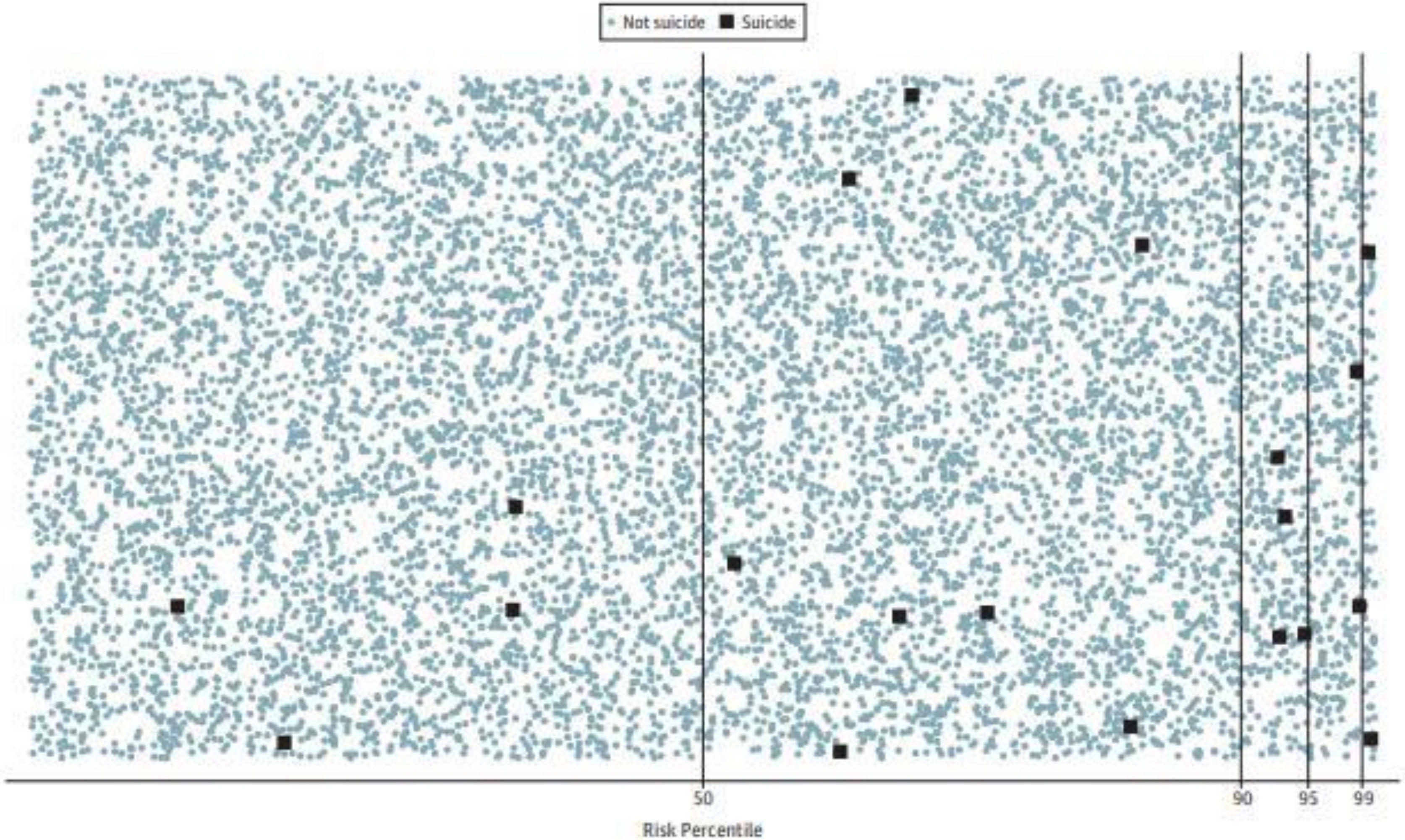
Prediction Models for Suicide Attempts and Deaths
A Systematic Review and Simulation

Bradley E. Belsher, PhD; Derek J. Smolenski, PhD, MPH; Larry D. Pruitt, PhD; Nigel E. Bush, PhD; Erin H. Beech, MA; Don E. Workman, PhD; Rebecca L. Morgan, PhD, MPH; Daniel P. Evatt, PhD; Jennifer Tucker, PhD; Nancy A. Skopp, PhD

Supplemental content

IMPORTANCE Suicide prediction models have the potential to improve the identification of

Figure 2. Illustration of Implementing a Suicide Prediction Model



Lezen van paper

- Lees de abstract, en bespreek het samen met je buurman

Key points

- This study developed machine learning algorithms to detect risk for suicide attempts among adolescents using only routinely collected clinical electronic health record data.
- By combining risk factors including comorbidities, medication usage, clinical encounter histories, socio-economic status, and demographics, machine learning produced accurate prediction across multiple cohort comparisons and time points.
- Applying machine learning to large and widely available clinical data may be a promising avenue toward scalable risk detection in the context of well-designed clinical decision support.

Background: Adolescents have high rates of nonfatal suicide attempts, but clinically practical risk prediction remains a challenge. Screening can be time consuming to implement at scale, if it is done at all. Computational algorithms may predict suicide risk using only routinely collected clinical data. We used a machine learning approach validated on longitudinal clinical data in adults to address this challenge in adolescents. **Methods:** This is a retrospective, longitudinal cohort study. Data were collected from the Vanderbilt Synthetic Derivative from January 1998 to December 2015 and included 974 adolescents with nonfatal suicide attempts and multiple control comparisons: 496 adolescents with other self-injury (OSI), 7,059 adolescents with depressive symptoms, and 25,081 adolescent general hospital controls. Candidate predictors included diagnostic, demographic, medication, and socioeconomic factors. Outcome was determined by multiexpert review of electronic health records. Random forests were validated with optimism adjustment at multiple time points (from 1 week to 2 years). Recalibration was done via isotonic regression. Evaluation metrics included discrimination (AUC, sensitivity/specificity, precision/recall) and calibration (calibration plots, slope/intercept, Brier score). **Results:** Computational models performed well and did not require face-to-face screening. Performance improved as suicide attempts became more imminent. Discrimination was good in comparison with OSI controls (AUC = 0.83 [0.82–0.84] at 720 days; AUC = 0.85 [0.84–0.87] at 7 days) and depressed controls (AUC = 0.87 [95% CI 0.85–0.90] at 720 days; 0.90 [0.85–0.94] at 7 days) and best in comparison with general hospital controls (AUC 0.94 [0.92–0.96] at 720 days; 0.97 [0.95–0.98] at 7 days). Random forests significantly outperformed logistic regression in every comparison. Recalibration improved performance as much as ninefold – clinical recommendations with poorly calibrated predictions can lead to decision errors. **Conclusions:** Machine learning on longitudinal clinical data may provide a scalable approach to broaden screening for risk of nonfatal suicide attempts in adolescents. **Keywords:** Suicide; attempted; adolescent; machine learning; decision support techniques; electronic health records.

Take home

- Big data en machine learning in de gezondheidszorg blijven
- Gezondheidszorg heeft andere uitdagingen dan het herkennen van een Cihuahua
- Verdiep je ook als behandelaar/onderzoeker ook in machine learning