Digital Technologies and Value Creation

Dr. Philippe Blaettchen
Bayes Business School (formerly Cass)

www.bayes.city.ac.uk

What we've done…

Building a solid foundation: Gathering data (internally, externally), cleaning, pre-processing

What we've done…

Description

General techniques, specific view on marketing and people analytics

Building a solid foundation: Gathering data (internally, externally), cleaning, pre-processing

… and where we are going

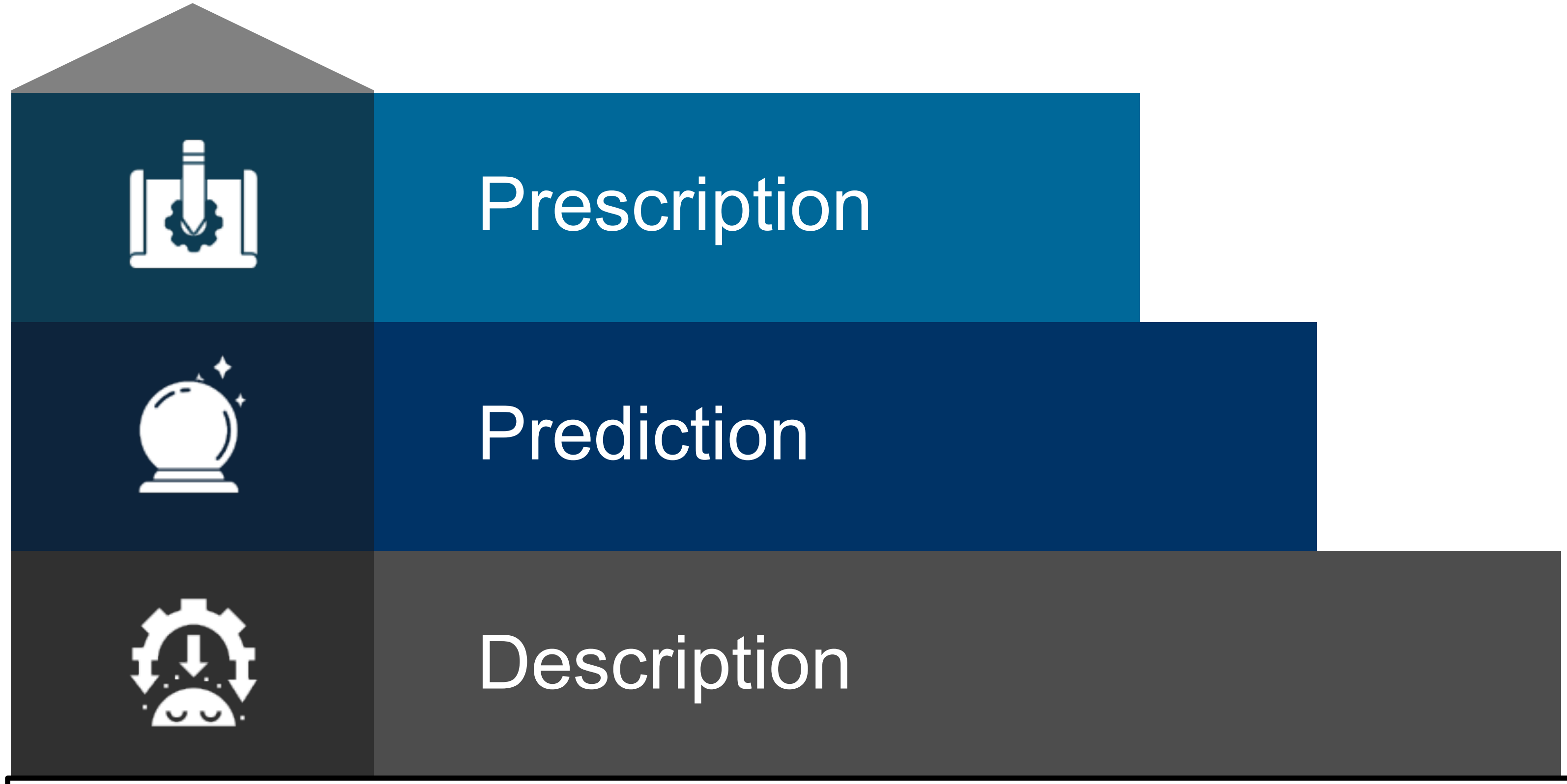
Supervised machine learning: regression and classification

Unsupervised machine learning: dimensionality reduction and clustering

Description

Building a solid foundation: Gathering data (internally, externally), cleaning, pre-processing

… and where we are going



Experimentation, optimization

Building a solid foundation: Gathering data (internally, externally), cleaning, pre-processing

Overview – subject to change

Overarching theme	Week	
Introduction	1	Introduction to analytics applications and coding basics
Gathering data	2	Scraping web data
Gathering data / descriptive analytics	3	Data pre-processing and descriptive analytics
Gathering data / descriptive analytics	4	Descriptives in marketing analytics, and using social media APIs
Descriptive analytics	5	Descriptives in people analytics
NO LECTURE	6	NO LECTURE
Predictive analytics	7	Retaining employees and customers with classification
Predictive analytics	8	Wrapping up classification and a deep-dive into dimensionality reduction
Predictive analytics	9	Segmenting customers and positioning products
Prescriptive analytics	10	Optimizing products and organizations
Prescriptive analytics	11	A/B-testing in practice

BAYES BUSINESS SCHOOL
CITY, UNIVERSITY OF LONDON

Learning objectives of today

Goals: Understand the difference between regression and classification

- Understand what logistic regression is, and why it is a key tool in the predictive analytics toolbox
- Learn some key metrics for evaluating a classification model

How will we do this?

- Recap the video materials
- Going back to Chimera Corp and its issues of high employee turnover

Project presentations

Video recap

Regression in the context of supervised machine learning

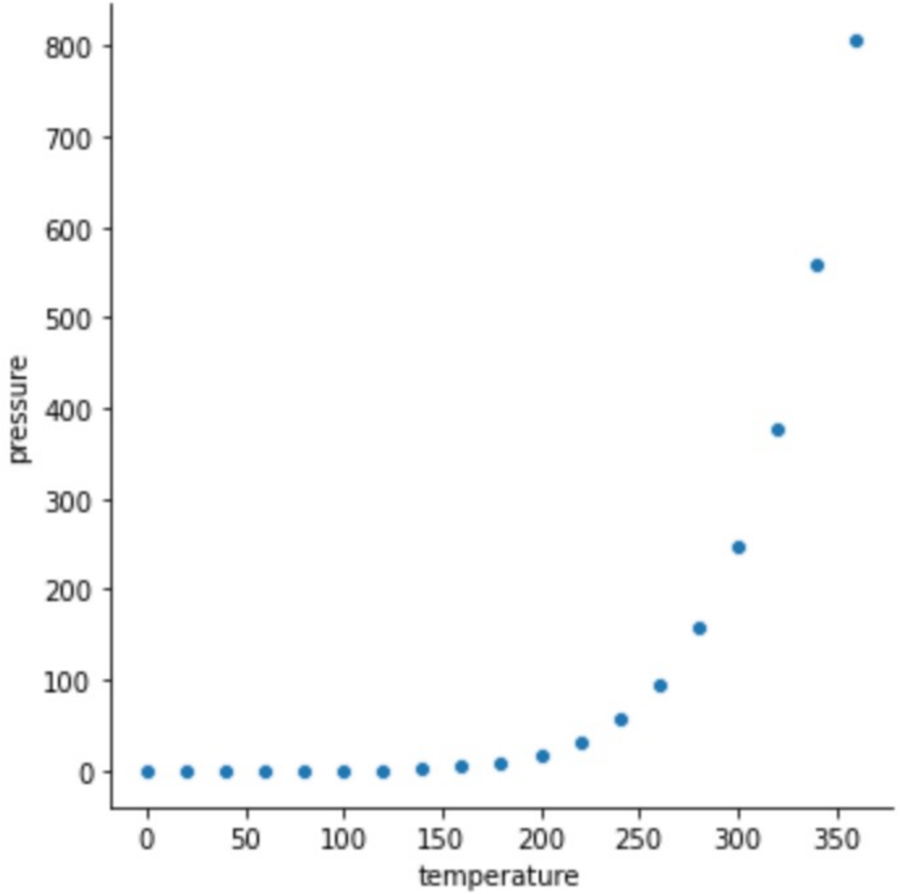
Why do we do linear regression (in a machine learning context)?

The goal is to predict new values.

For example, if height is x we should be able to infer weight y by using our regression line

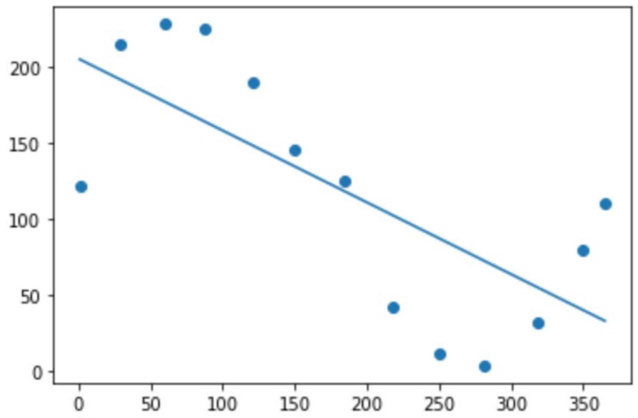
Linear regression is not always good enough…

Many relationships are simply not linear:

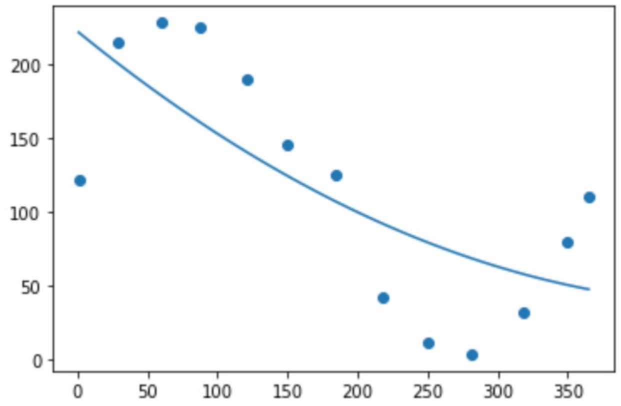


BAYES BUSINESS SCHOOL
CITY, UNIVERSITY OF LONDON

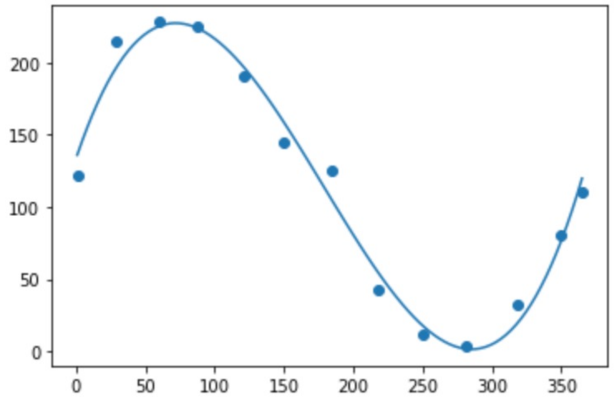
Predicting sales using polynomial regression and the problem of overfitting



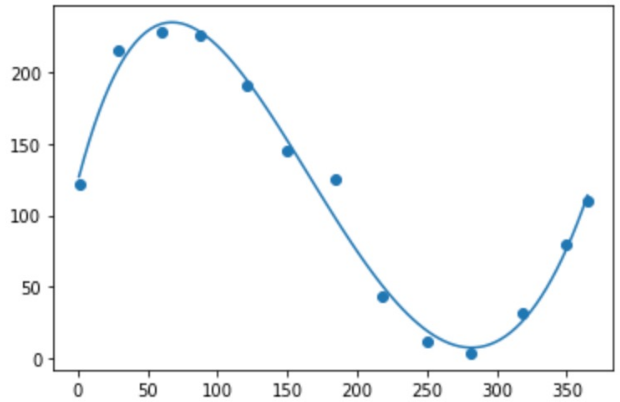
d=1, R^2=0.516



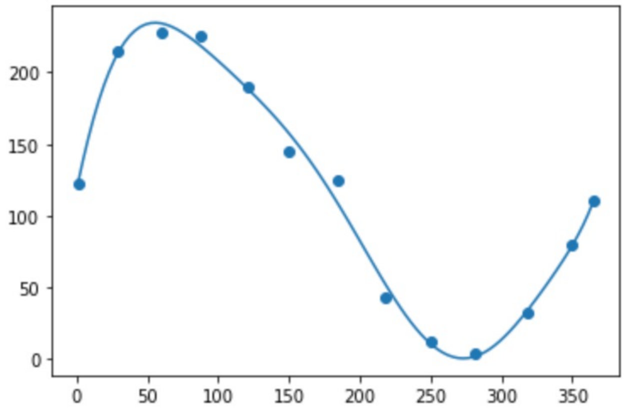
d=2, R^2=0.531



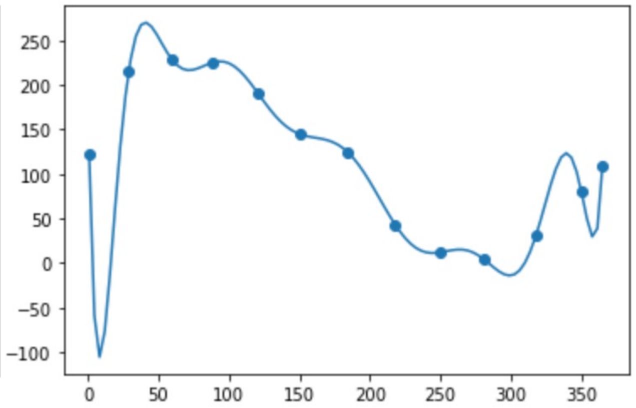
d=3, R^2=0.979



d=5, R^2=0.985



d=8, R^2=0.992



d=12, R^2=1

Validation set

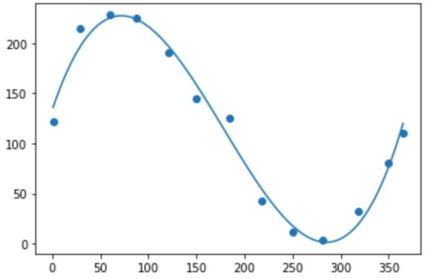
- Not good enough to have a regression that works well **just on the data we have at hand**
- Need to know whether it will perform well **for new data as well → training/testing split**
- Additionally, we can use validation to choose between models

BAYES
BUSINESS SCHOOL
CITY, UNIVERSITY OF LONDON

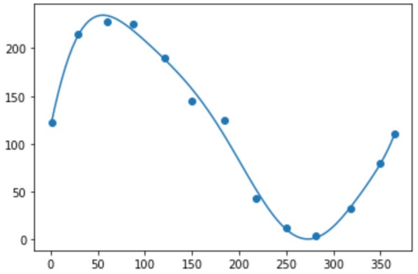
Training, test, and validation sets in practice

Example: Polynomial regression

1. Use the **training set** to come up with polynomial regressors with different degrees.



d=3



d=8

2. Use the **validation set** to pick which degree is the best, e.g., $d = 3$.

3. Use the **testing set** to evaluate how well the model you picked would perform on new data.

From regression to classification

A question to get started

Consider these situations:

- A transaction has just been made on a customer's card. Using the user's past transactions and the location of the transaction, can I determine whether the transaction was fraudulent or not?
- A patient has just turned up at the hospital I work in. (S)he has a set of symptoms. Can I determine whether the patient is suffering from meningitis or covid or the flu?

Think about the following question:

1. **Would regression work to answer one, both or none of these questions?**
2. **Why/why not?**

BAYES
BUSINESS SCHOOL
CITY, UNIVERSITY OF LONDON

Regression

- **Input :** (x_i, y_i) with x_i =features and y_i =response
- **Goal:** given x_i, predict y_i
- Key fact: y_i here is a **number**

Classification

- **Input :** (x_i, y_i) with x_i =features and y_i =response
- **Goal:** given x_i, predict y_i
- Key fact: y_i here is a **categorical variable**

  yes or no: binary classification

  meningitis or flu or covid: multinomial classification

BAYES BUSINESS SCHOOL
CITY, UNIVERSITY OF LONDON

Regression vs. Classification in practice

Supervised learning

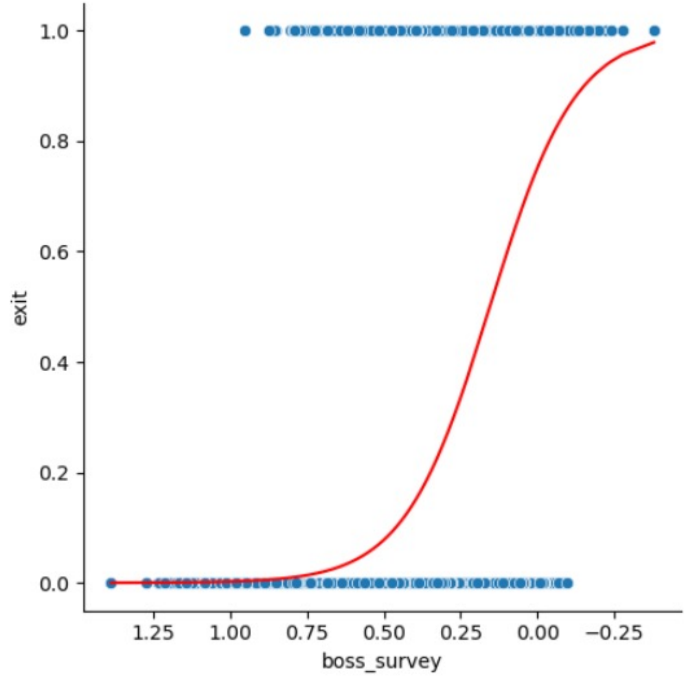
- **Regression:** predict a **numerical** value
  - Linear/polynomial regression
  - Tree-based methods, k-NN, spline methods
- **Classification:** predict a **categorical** value
  - Logistic regression
  - Tree-based methods, SVM, k-NN, LDA, QDA

Our use-case

- It has about **18,000** employees

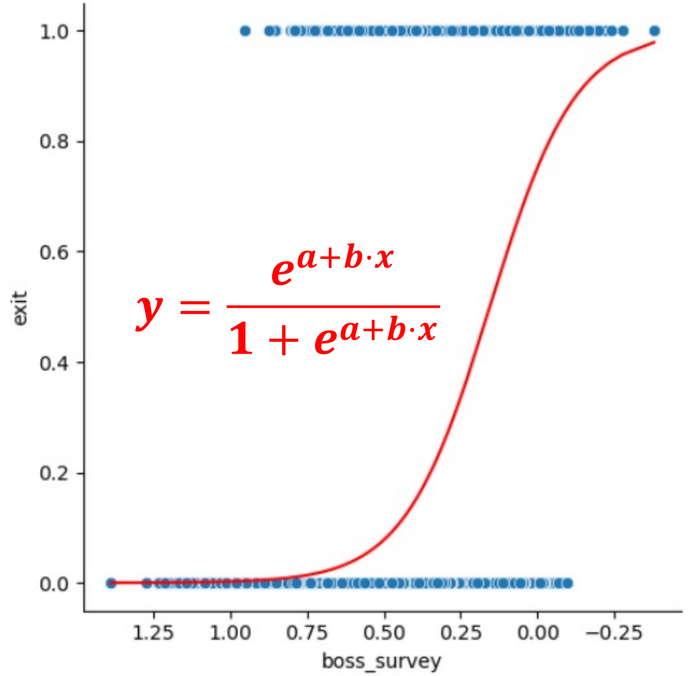
- The attrition rate of employees is quite high (**~14%** compared to industry average of **8-9%**).
  → How to fix it?
  - They have gathered data for calendar year 2020

What happens in logistic regression?



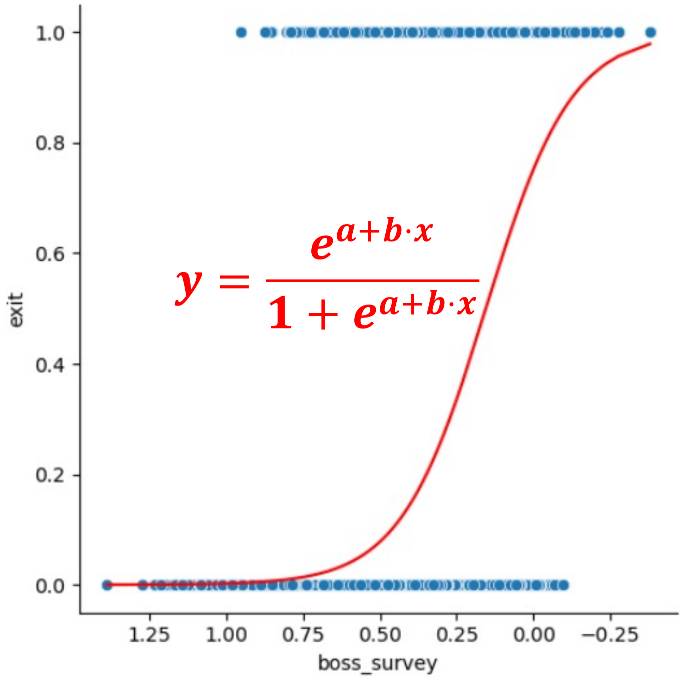
- **Input:** datapoints (x_i, y_i)
- Here x_i is the boss survey result; $y_i = 1$ (exit) or 0 (doesn't exit)
- We have around 18,000 datapoints.

What happens in logistic regression?



- **Goal:** Find numbers (a, b) such that
- $\frac{e^{a+b\cdot x_i}}{1+e^{a+b\cdot x_i}}$ is as close as possible to y_i for all 18,000 observations

Logistic regression



Advantages:

$$y = \frac{e^{a+b\cdot x}}{1+e^{a+b\cdot x}}$$

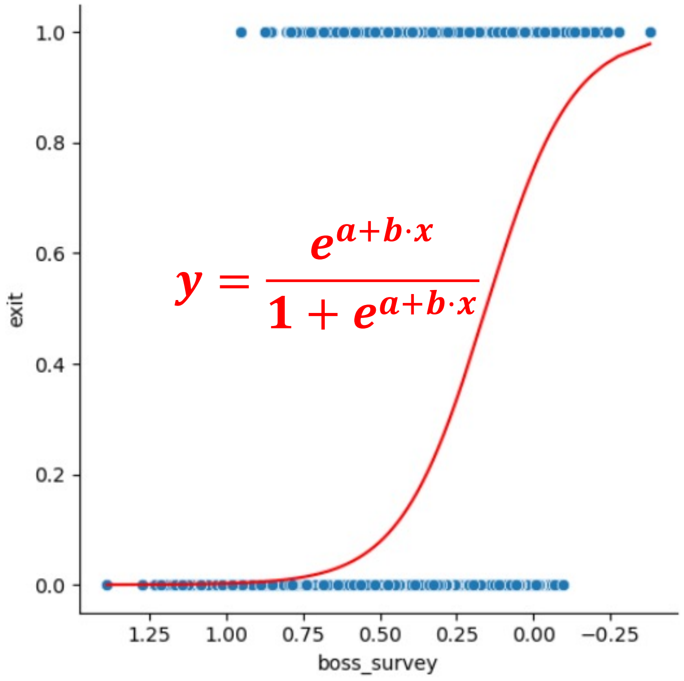
is a number between 0 and 1 (why?).

Output: The predicted value

$$y_{pred_i} = \frac{e^{a+b\cdot x_i}}{1+e^{a+b\cdot x_i}}$$

is always between 0 and 1 and can be interpreted as the **probability that observation x_i exits (i.e., $P(y_i = 1)$).**

Logistic regression

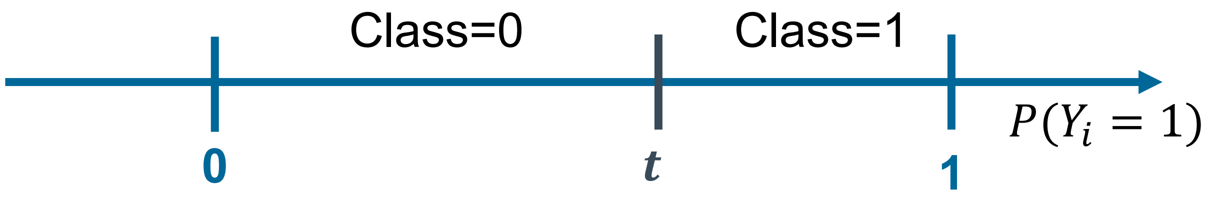


For each x_i,
we get a **probability** that the corresponding employee **will exit**.

How to go from there to **classification**? i.e.,

How to **decide on labels** (exits/doesn't exit) to give to the employee?

Key idea: threshold t (by default in scikit 50%)



Logistic regression in Python (using scikit)

Threshold setting and performance metrics

We still use the training/validation/testing paradigm

- Can also still use cross-validation. The supervised learning process is the same.
- Only difference: **what metrics to use on the testing set?**
- Saw MAPE and RMSE for regression. What can we use for logistic regression?

Please don't be too confused by the confusion matrix

	Predicted class=0 (i.e., predicted stays)	Predicted class=1 (i.e., predicted exit)
Actual class=0 (i.e., actual stays)	True Negatives (TN)	False Positives (FP)
Actual class=1 (i.e., actual exits)	False Negatives (FN)	True Positives (TP)

Specificity $= \frac{TN}{FP+TN}$ (proportion of true negatives correctly identified)

Sensitivity $= \frac{TP}{TP+FN}$ (proportion of true positives correctly identified)

Accuracy = $\frac{TP+TN}{TP+TN+FP+FN}$, Misclassification =$\frac{FP+FN}{TP+TN+FP+FN}$

BAYES BUSINESS SCHOOL
CITY, UNIVERSITY OF LONDON

Setting the threshold

- Can view **threshold t as a hyperparameter**
- Use **validation set** to fix threshold appropriately

(1) Train logistic model: obtain probabilities $P(y_i = 1)$

(2) Use validation set to pick right threshold

(4) Test the accuracy of your model on the testing set

(3) Retrain your model on training + validation. Using the threshold found, classify your observations.

How to do (2)?

BAYES BUSINESS SCHOOL
CITY, UNIVERSITY OF LONDON

Setting the threshold

- Training gives us **the coefficients** $\boldsymbol{a, b_1, \dots, b_n}$ such that

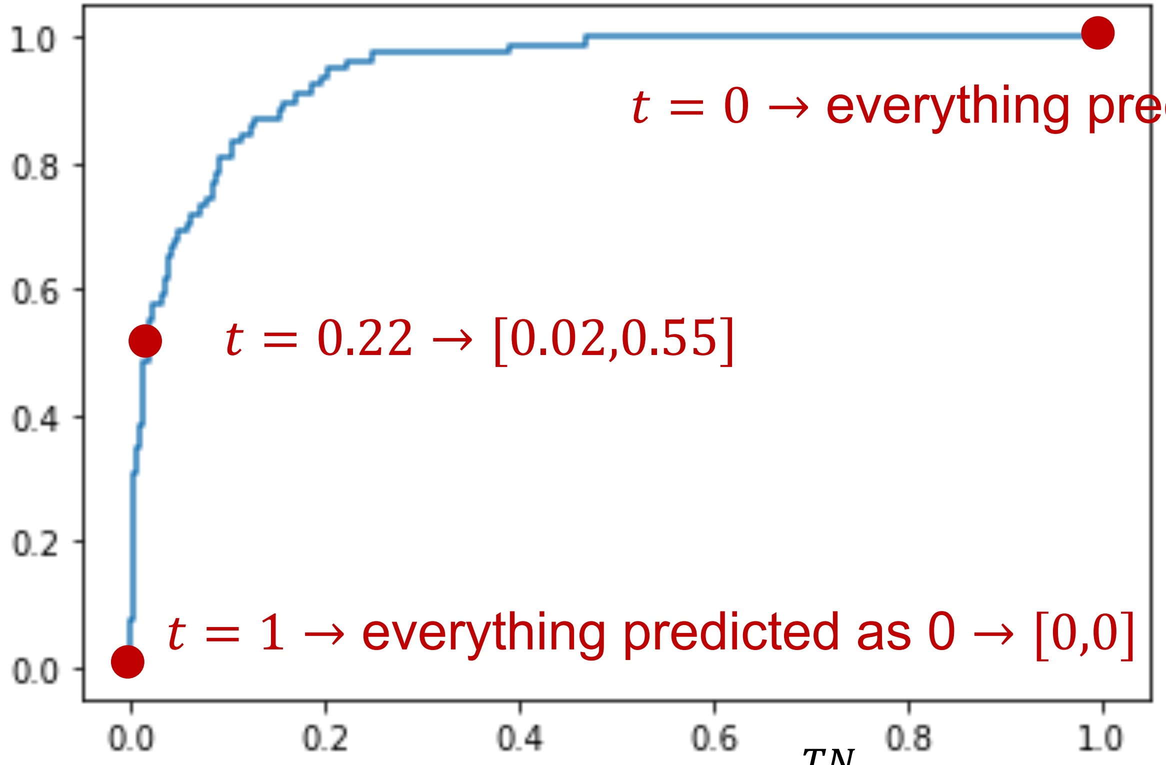
$$P(y_i = 1) = \frac{e^{a+b_1 \cdot x_i^1 + b_2 x_i^2 + \cdots + b_n x_i^n}}{1 + e^{a+b_1 \cdot x_i^1 + b_2 x_i^2 + \cdots + b_n x_i^n}}$$

- From the validation set, we can obtain $\boldsymbol{P(y_i = 1)}$ **for any observation** $\boldsymbol{x_i}$ in the **validation set** using the equation above.

- We can then pick different thresholds and for each threshold, compute the number of **FP/TP/FN/TN** that we get on the **validation set** when that **threshold is picked.**

- Hence, **for each threshold** $\boldsymbol{t}$, **we can obtain [sensitivity, specificity]** over the validation set: this gives the ROC curve.

Sensitivity $= \frac{TP}{TP+FN}$ (true positive rate)



$t = 0 \rightarrow$ everything predicted as 1 $\rightarrow [1,1]$

$t = 0.22 \rightarrow [0.02, 0.55]$

$t = 1 \rightarrow$ everything predicted as 0 $\rightarrow [0,0]$

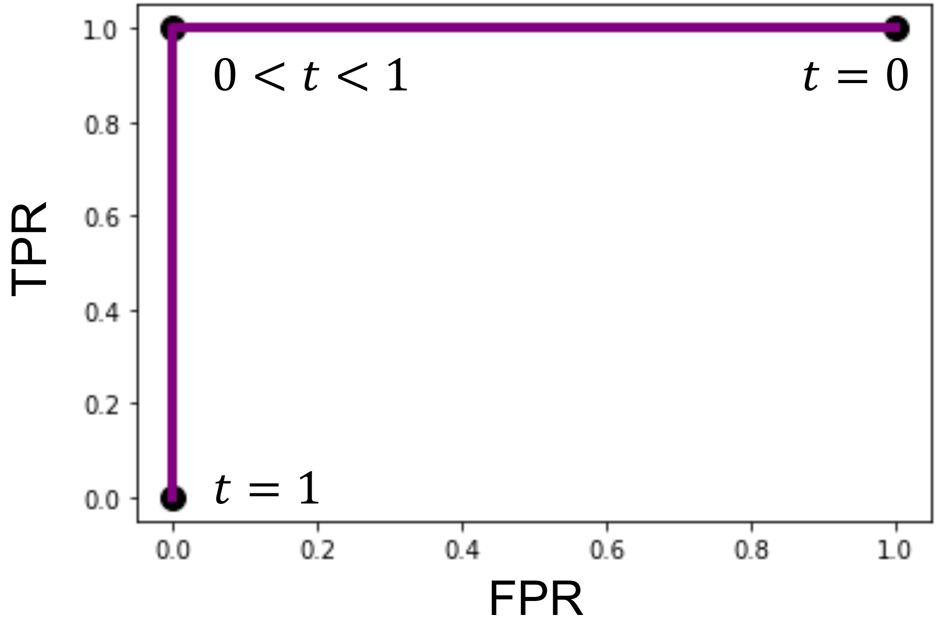
1 $-$specificity$=1 - \frac{TN}{FP+TN}$ (false positive rate)

As t increases: more of the observations are predicted to 0 $\Rightarrow FN \uparrow, TN \uparrow, FP \downarrow, TP \downarrow$
$\Rightarrow$ Sensitivity $\downarrow 0$ and 1-specificity $\downarrow 0$

BAYES BUSINESS SCHOOL
CITY, UNIVERSITY OF LONDON

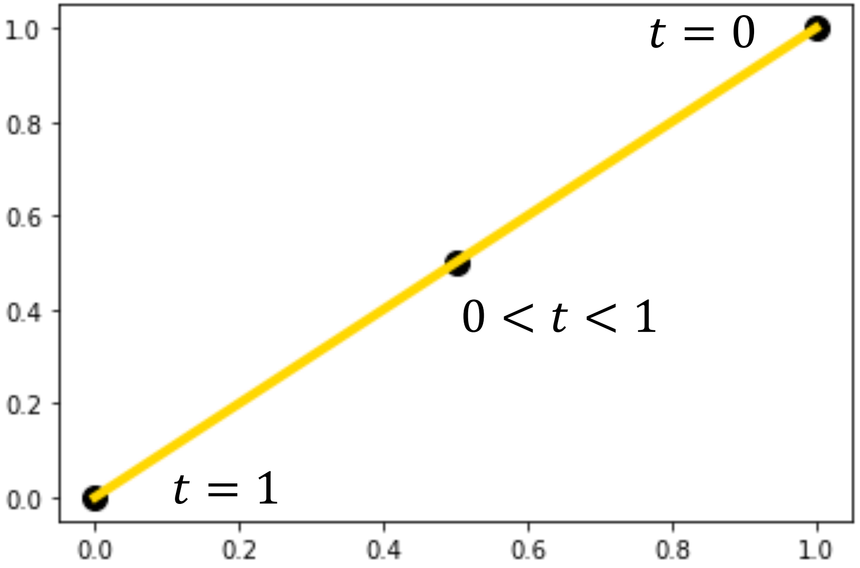
Good and bad ROC curves

Best ROC curve



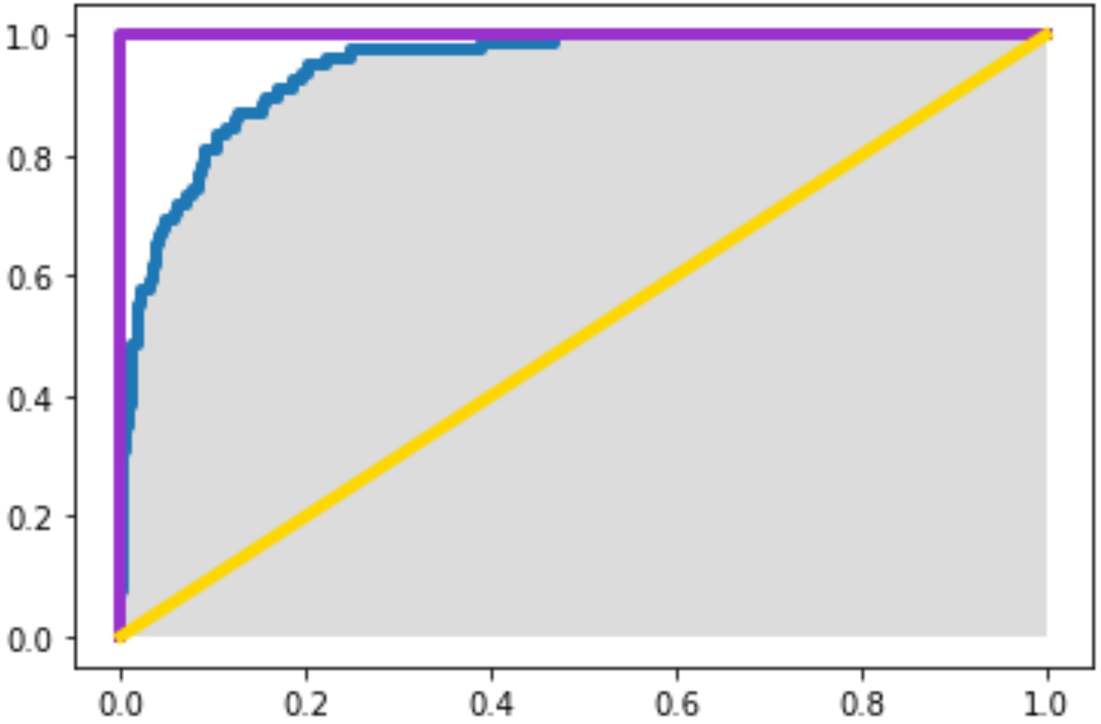
When $0 < t < 1$: predict perfectly,
i.e., TPR=1, FPR=0.

Worst ROC curve



When $0 < t < 1$: predict at random:
→ TPR=1/2, FPR=1/2

Summarizing the ROC curve: AUC



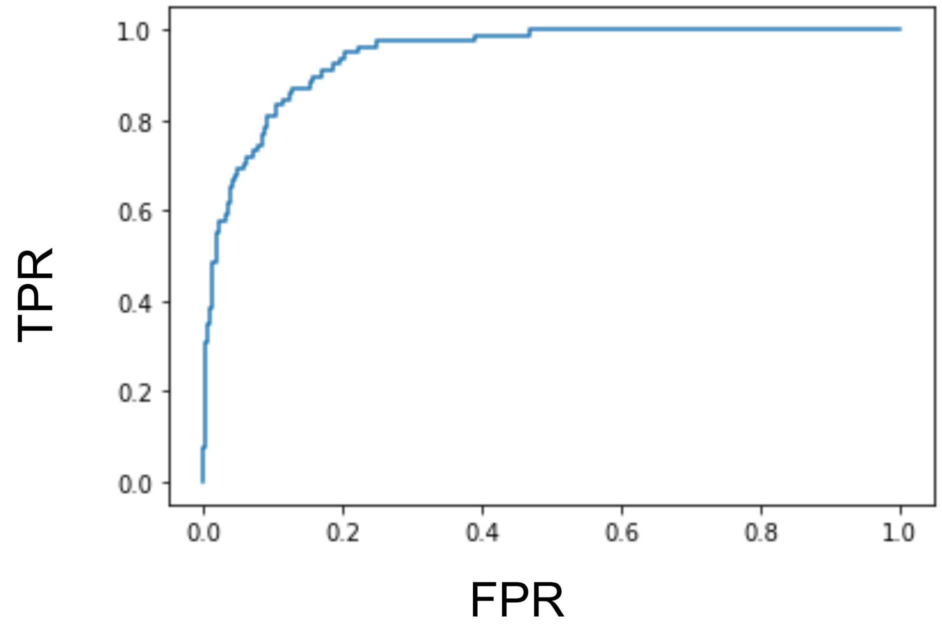
AUC	Quality of Prediction
0.50	Random
0.50-0.60	Fail
0.60-0.70	Poor
0.70-0.80	Fair
0.80-0.90	Good
0.90-1	Excellent

*context specific (driverless car vs fin. instrument)

- **Gray area**: area under the curve (AUC)
- Measures how good the model is before picking a threshold
- **Interpretation**: given a random positive and negative, amount of time their classes are correctly predicted

Back to setting the threshold

- Trade-off between False Positives and False Negatives.
- Pick **best threshold for best trade-off**
- **Evaluate which is best for the application:** raising false alarms or failing to detect positives
- In our default case: exits are costly – we prefer to “over-target” employees rather than lose them.
- FP high: ok, FN need to be low → need TPR close to 1, ok if FPR not exactly 0.



Setting the threshold in practice

python™

BAYES
BUSINESS SCHOOL
CITY, UNIVERSITY OF LONDON

How good is this model?

- Re-**train** the model on **Training+Validation**
- Apply the model to **test set** to get **predicted probabilities**
- Use the **0.1358 threshold** to classify each predicted probability to a **class**
- Obtain a **confusion matrix & accuracy measure**

Correctly avoided targeting 2650 employees who stayed

	Predicted to not exit	Predicted to exit
Did not exit	2650	1237
Did exit	402	208

Possibly wasted efforts on 1237 employees

Missed 402 employees who should have been targeted

Correctly targeted 208 employees who would exit otherwise

$$\textbf{Accuracy} = \frac{2650+208}{2650+1237+402+208} = 64\%$$

BAYES BUSINESS SCHOOL
CITY, UNIVERSITY OF LONDON

Final remarks

Importance of classification

- People analytics: predicting behaviors of employees
  - E.g., staying/leaving, making a promotion, accepting a job offer, managing a task successfully

- Marketing analytics: predicting behaviors of customers
  - E.g., buying/not buying, upgrading, returning

- In either case, organizations treat people differently based on predicted behaviors - targeted strategies
  - Discrete outcomes easier to observe and act upon!

Predicting more than two outcomes

- Logistic regression as we presented it enables us to **answer yes/no questions**
- Think back to example: how to predict whether it is meningitis/flu/covid?
- This requires more than yes/no: requires us to predict **one of three categories**
- Can be done via an adaptation of logistic regression (can you maybe see how we could generalize the binary model?): called **multinomial logistic regression**
- Not covered here

See you next week!