1) Introduction
→ Targeted marketing example

2) Intro. to predictive modeling
→ regression

Introduction to Data Analytics & Applications

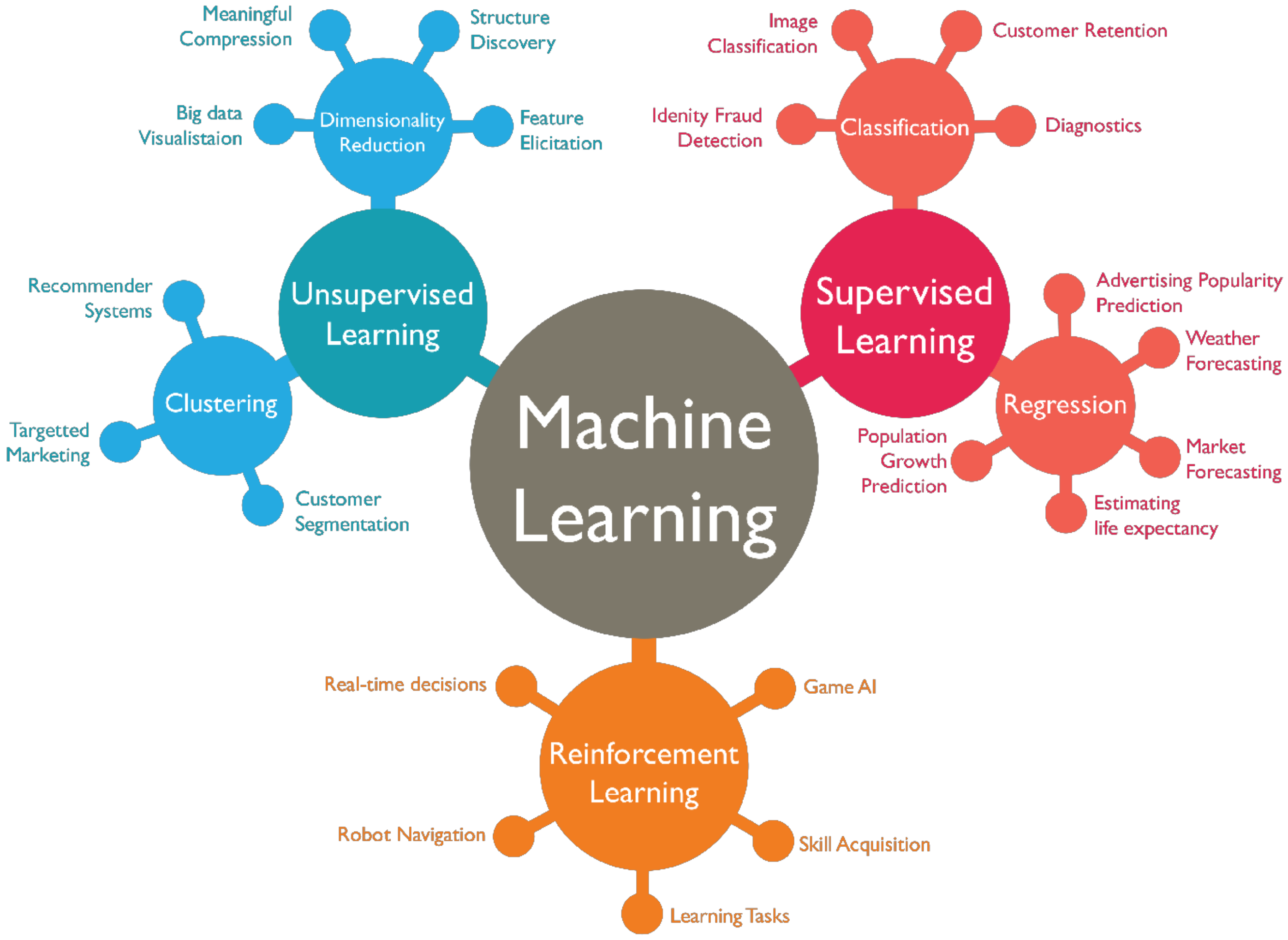
DUKE
FUQUA
SCHOOL OF BUSINESS

Goals for Data Analytics & Applications

- Main goals:
    - To introduce you to some tools for extracting useful knowledge from data
    - To develop literacy with basic concepts and terminology from data science
    - To practice the art and science of “data-analytic” thinking in the context of real-world problems

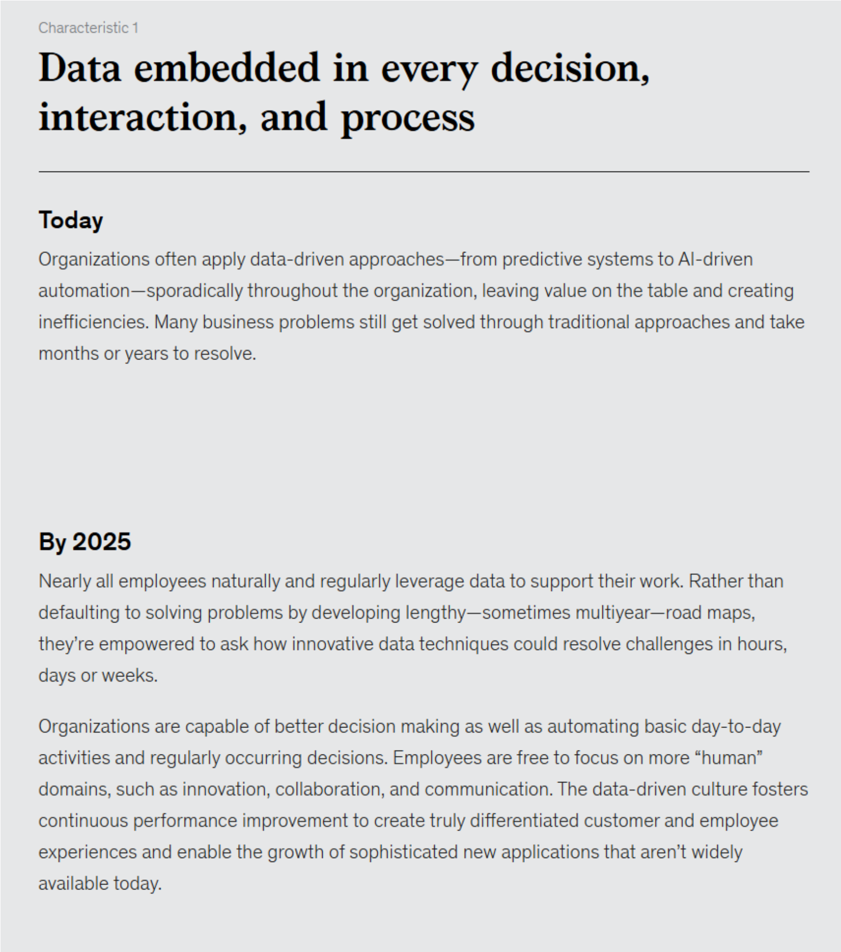
- Also:
    - A data-analytic mindset provides a *conceptual framework* in addition to an understanding of computational methods
    - Some practice with Python: many other tools available

A web diagram of data analytics



Source: https://www.kdnuggets.com/2019/04/poll-data-science-machine-learning-methods-algorithms-use-2018-2019.html

The value and ubiquity of data analytics



Characteristic 1

Data embedded in every decision, interaction, and process

Today

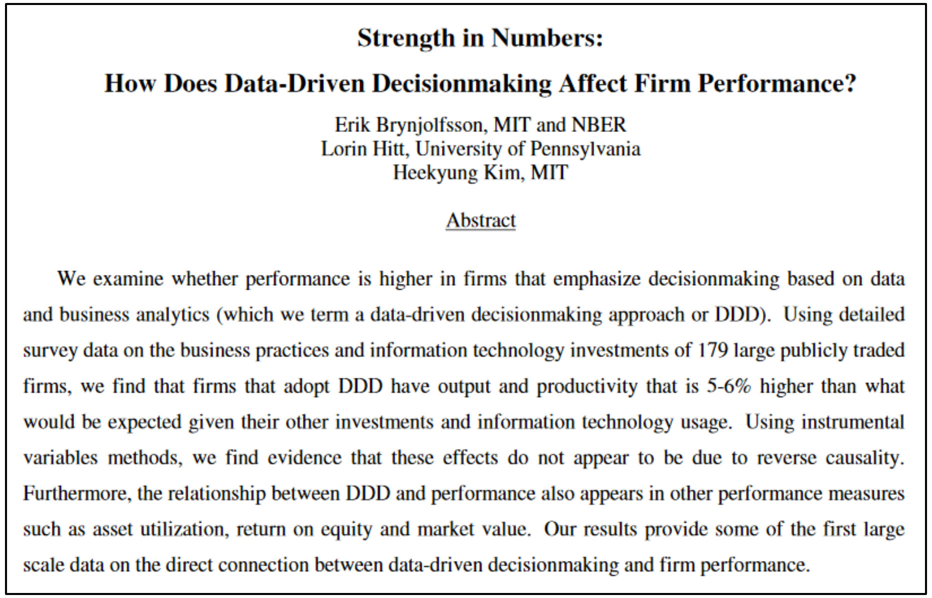
Organizations often apply data-driven approaches—from predictive systems to AI-driven automation—sporadically throughout the organization, leaving value on the table and creating inefficiencies. Many business problems still get solved through traditional approaches and take months or years to resolve.

By 2025

Nearly all employees naturally and regularly leverage data to support their work. Rather than defaulting to solving problems by developing lengthy—sometimes multiyear—road maps, they're empowered to ask how innovative data techniques could resolve challenges in hours, days or weeks.

Organizations are capable of better decision making as well as automating basic day-to-day activities and regularly occurring decisions. Employees are free to focus on more "human" domains, such as innovation, collaboration, and communication. The data-driven culture fosters continuous performance improvement to create truly differentiated customer and employee experiences and enable the growth of sophisticated new applications that aren't widely available today.

"The data-driven enterprise of 2025," https://www.mckinsey.com



Strength in Numbers:

How Does Data-Driven Decisionmaking Affect Firm Performance?

Erik Brynjolfsson, MIT and NBER
Lorin Hitt, University of Pennsylvania
Heekyung Kim, MIT

Abstract

We examine whether performance is higher in firms that emphasize decisionmaking based on data and business analytics (which we term a data-driven decisionmaking approach or DDD). Using detailed survey data on the business practices and information technology investments of 179 large publicly traded firms, we find that firms that adopt DDD have output and productivity that is 5-6% higher than what would be expected given their other investments and information technology usage. Using instrumental variables methods, we find evidence that these effects do not appear to be due to reverse causality. Furthermore, the relationship between DDD and performance also appears in other performance measures such as asset utilization, return on equity and market value. Our results provide some of the first large scale data on the direct connection between data-driven decisionmaking and firm performance.

Brynjolfsson et al. (2011), https://papers.ssrn.com/sol3/papers.cfm?abstract_id=1819486

GLOBAL
BIG DATA ANALYTICS
IN HEALTHCARE
MARKET
OPPORTUNITIES AND FORECASTS, 2018-2025

Global Big Data Analytics in Healthcare Market is expected to reach $67.82 Billion by 2025.

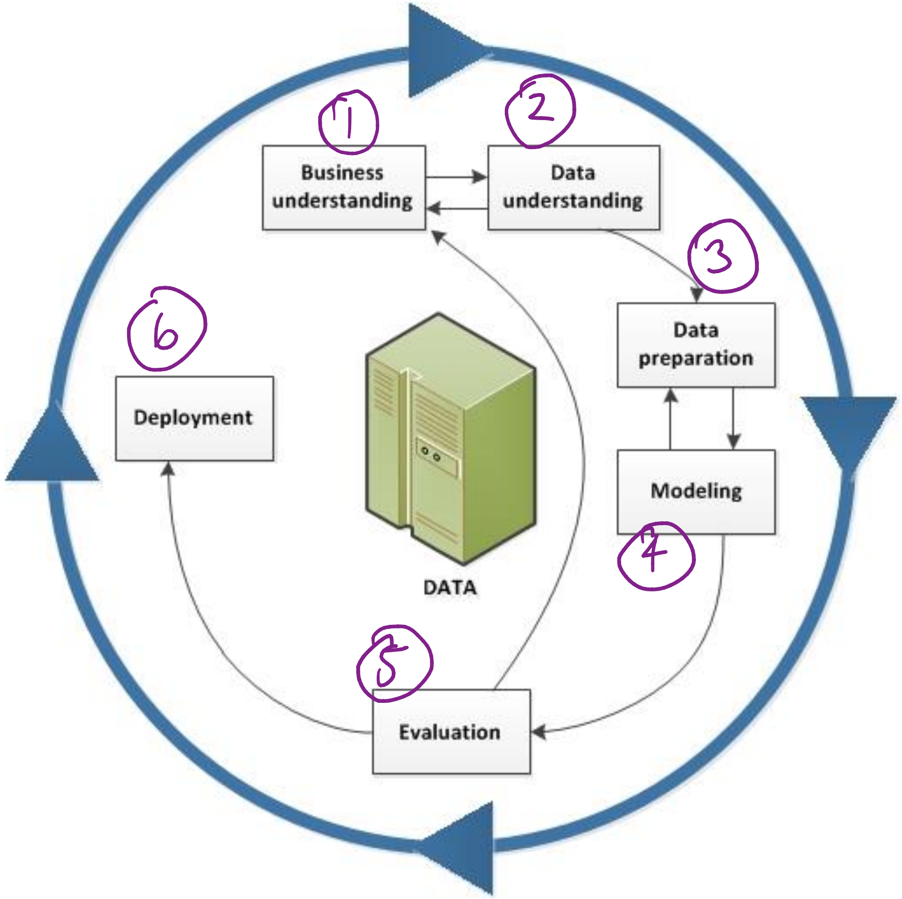
Growing at a CAGR of 19.10% (2018-2025)

BIG
DATA
IOT
©Allied Market Research

https://www.einnews.com

Organizational view of data analytics

CRISP-DM: Cross-Industry Standard Process for Data Mining



“The data mining life cycle.” Source: ibm.com

Some fundamental principles

1. Although every problem differs, there is a systematic process for using data to help make better decisions.

2. All data contains structure, but we need to distinguish between the “signal” and the “noise.”

3. The specifics of the problem we are trying to solve should govern the choice of solution techniques, not the other way around.

4. Data and data analytics (or data science) capabilities are strategic assets, and firms and organizations need to carefully consider their investments in these assets.

Targeted marketing example

Your company offers subscription-based services to customers in a market with many competitors, and you are concerned about retaining customers at the expiration of their contracts. You are considering offering incentives (e.g., discounts on service fees) to some customers near the end of their contracts to retain them. To which customers should you offer incentives?

① What are the key uncertainties?

$$P(\underset{\text{stays}}{\underset{\uparrow}{S}} \mid \underset{\text{features}}{\underset{\uparrow}{x}}, \text{target}) = \text{probability customer } x \text{ stays if we target}$$

$$P(S \mid x, \text{not target}) = \text{'' \quad '' \quad '' \quad '' if we do not target}$$

② What is our objective? Maximize expected profits

(Additional space for notes)

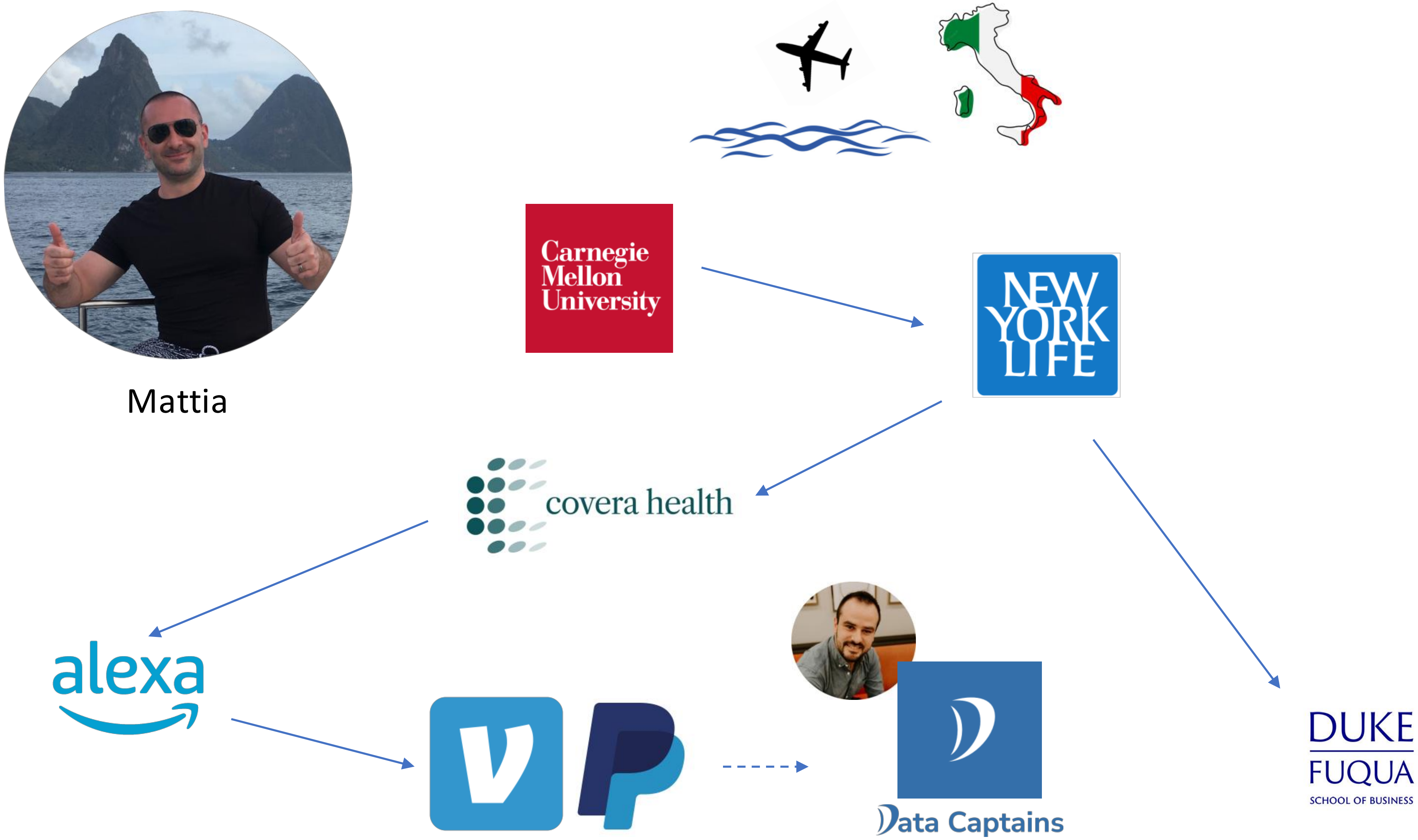
	Target	Not Target
Stay	$V_s(x) - c$	$V_s(x)$
Not	$-c$	0

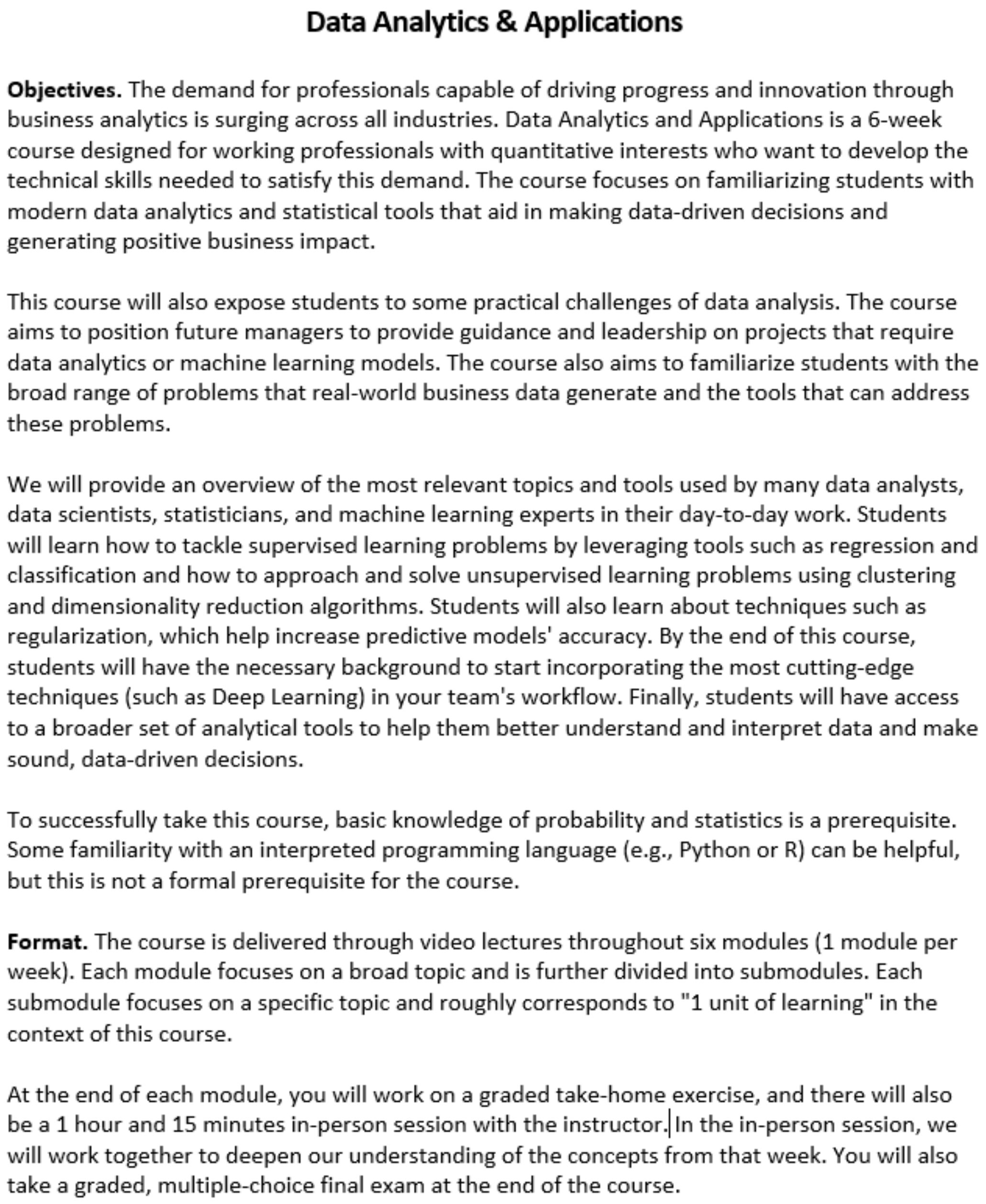
$$EV(\text{target}) = P(S|x, \text{target}) \cdot (V_s(x) - c) + (1 - P(S|x, \text{target})) \cdot (-c)$$

$$- \; EV(\text{not target}) = P(S|x, \text{not target}) \cdot V_s(x)$$

$$\Delta \text{Value}(\text{target}) = \left[P(S|x, \text{target}) - P(S|x, \text{not target})\right] \cdot V_s(x) - c$$

Your (video) instructor: Mattia Ciollaro





Data Analytics & Applications

Objectives. The demand for professionals capable of driving progress and innovation through business analytics is surging across all industries. Data Analytics and Applications is a 6-week course designed for working professionals with quantitative interests who want to develop the technical skills needed to satisfy this demand. The course focuses on familiarizing students with modern data analytics and statistical tools that aid in making data-driven decisions and generating positive business impact.

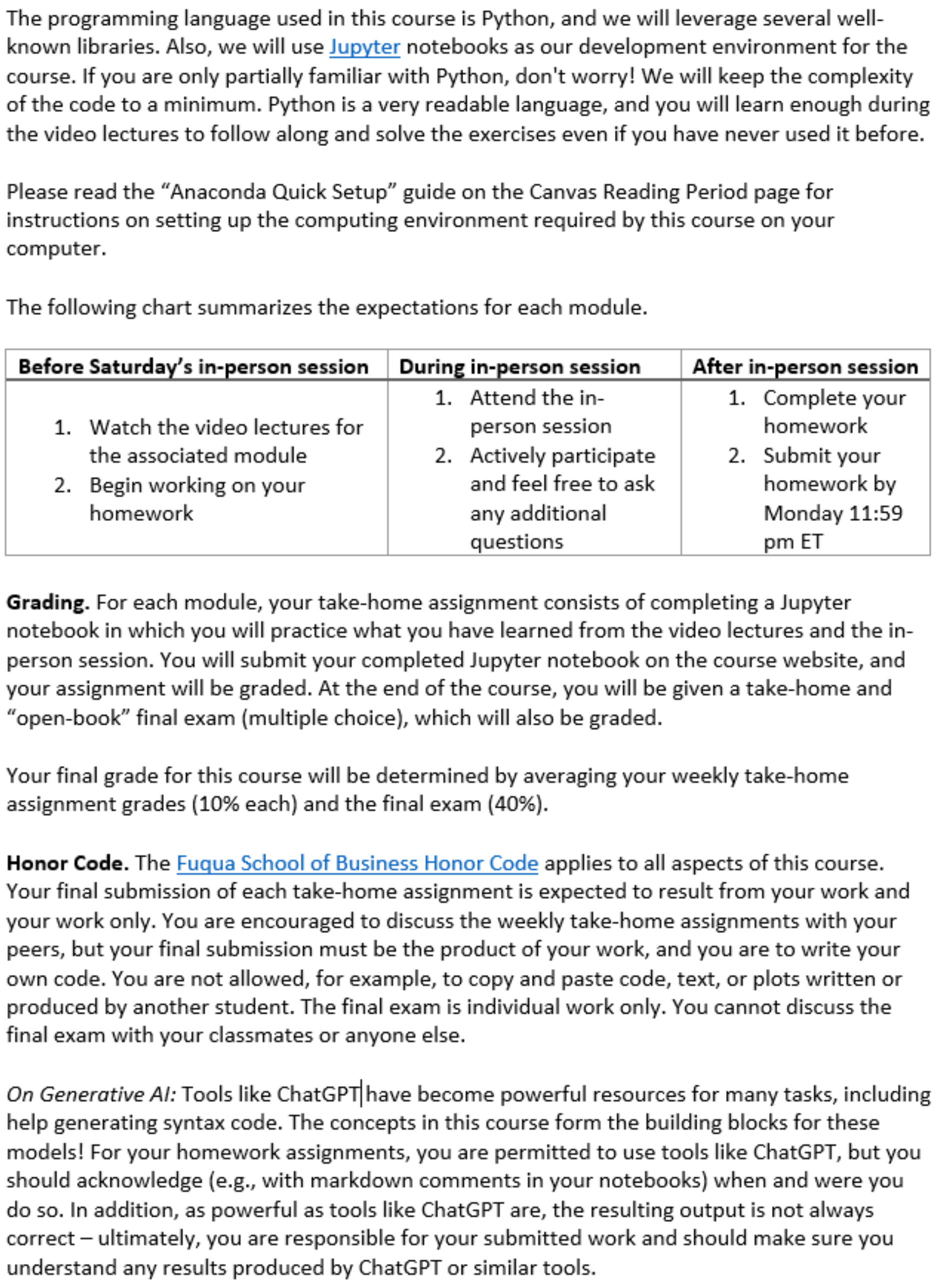
This course will also expose students to some practical challenges of data analysis. The course aims to position future managers to provide guidance and leadership on projects that require data analytics or machine learning models. The course also aims to familiarize students with the broad range of problems that real-world business data generate and the tools that can address these problems.

We will provide an overview of the most relevant topics and tools used by many data analysts, data scientists, statisticians, and machine learning experts in their day-to-day work. Students will learn how to tackle supervised learning problems by leveraging tools such as regression and classification and how to approach and solve unsupervised learning problems using clustering and dimensionality reduction algorithms. Students will also learn about techniques such as regularization, which help increase predictive models' accuracy. By the end of this course, students will have the necessary background to start incorporating the most cutting-edge techniques (such as Deep Learning) in your team's workflow. Finally, students will have access to a broader set of analytical tools to help them better understand and interpret data and make sound, data-driven decisions.

To successfully take this course, basic knowledge of probability and statistics is a prerequisite. Some familiarity with an interpreted programming language (e.g., Python or R) can be helpful, but this is not a formal prerequisite for the course.

Format. The course is delivered through video lectures throughout six modules (1 module per week). Each module focuses on a broad topic and is further divided into submodules. Each submodule focuses on a specific topic and roughly corresponds to "1 unit of learning" in the context of this course.

At the end of each module, you will work on a graded take-home exercise, and there will also be a 1 hour and 15 minutes in-person session with the instructor. In the in-person session, we will work together to deepen our understanding of the concepts from that week. You will also take a graded, multiple-choice final exam at the end of the course.



The programming language used in this course is Python, and we will leverage several well-known libraries. Also, we will use Jupyter notebooks as our development environment for the course. If you are only partially familiar with Python, don't worry! We will keep the complexity of the code to a minimum. Python is a very readable language, and you will learn enough during the video lectures to follow along and solve the exercises even if you have never used it before.

Please read the "Anaconda Quick Setup" guide on the Canvas Reading Period page for instructions on setting up the computing environment required by this course on your computer.

The following chart summarizes the expectations for each module.

Before Saturday's in-person session	During in-person session	After in-person session
1. Watch the video lectures for the associated module 2. Begin working on your homework	1. Attend the in-person session 2. Actively participate and feel free to ask any additional questions	1. Complete your homework 2. Submit your homework by Monday 11:59 pm ET

Grading. For each module, your take-home assignment consists of completing a Jupyter notebook in which you will practice what you have learned from the video lectures and the in-person session. You will submit your completed Jupyter notebook on the course website, and your assignment will be graded. At the end of the course, you will be given a take-home and "open-book" final exam (multiple choice), which will also be graded.

Your final grade for this course will be determined by averaging your weekly take-home assignment grades (10% each) and the final exam (40%).

Honor Code. The Fuqua School of Business Honor Code applies to all aspects of this course. Your final submission of each take-home assignment is expected to result from your work and your work only. You are encouraged to discuss the weekly take-home assignments with your peers, but your final submission must be the product of your work, and you are to write your own code. You are not allowed, for example, to copy and paste code, text, or plots written or produced by another student. The final exam is individual work only. You cannot discuss the final exam with your classmates or anyone else.

On Generative AI: Tools like ChatGPT have become powerful resources for many tasks, including help generating syntax code. The concepts in this course form the building blocks for these models! For your homework assignments, you are permitted to use tools like ChatGPT, but you should acknowledge (e.g., with markdown comments in your notebooks) when and were you do so. In addition, as powerful as tools like ChatGPT are, the resulting output is not always correct – ultimately, you are responsible for your submitted work and should make sure you understand any results produced by ChatGPT or similar tools.

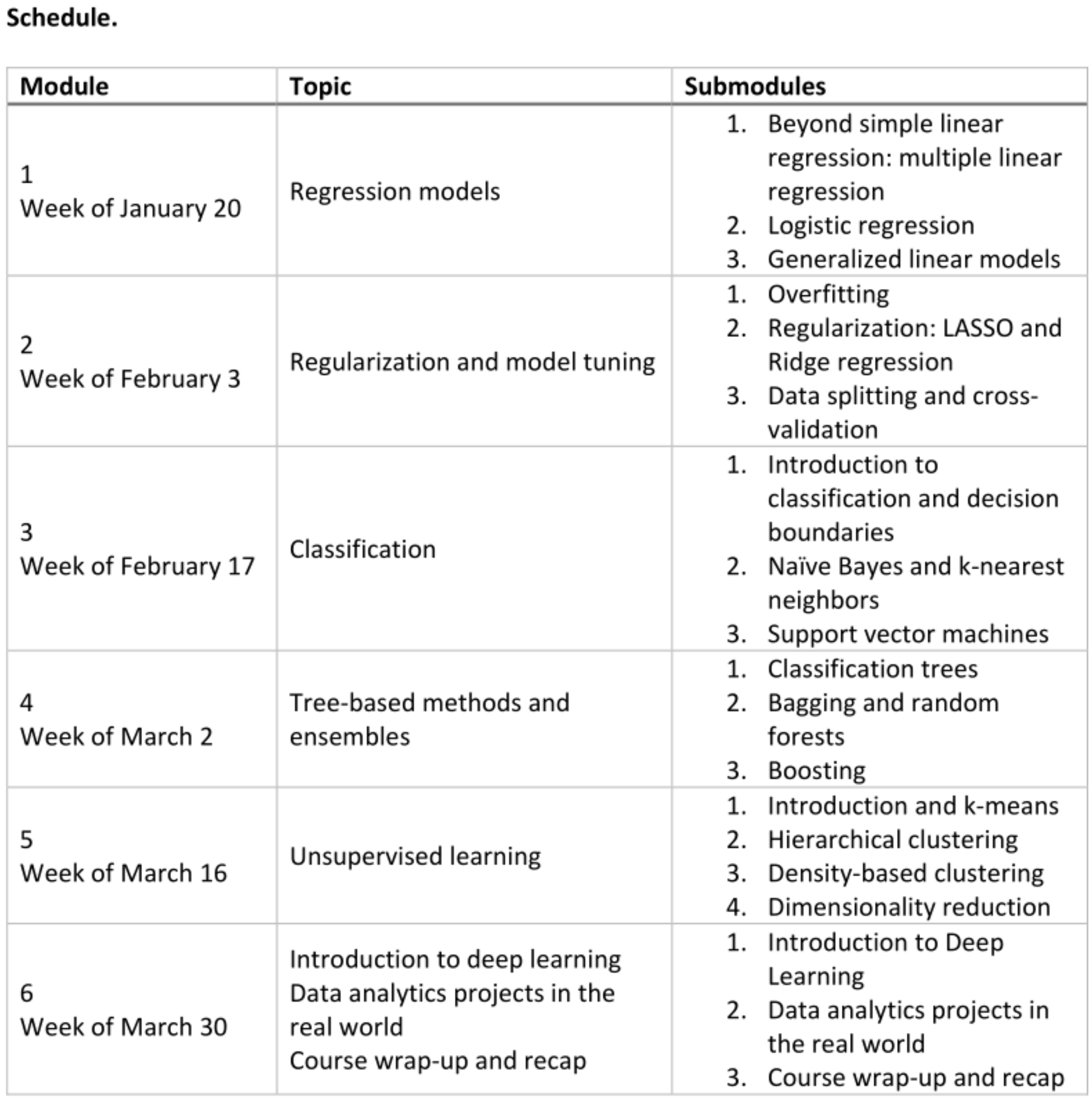
Supplementary Resources. The course is self-contained, and there are no required books. However, you may find some of these references useful should you decide to dive deeper into some of the topics covered in the video lectures:

- A Concise Introduction to Programming in Python, by Mark J. Johnson (beginner)
- Learning Python, by Mark Lutz (intermediate)
- Applied Predictive Modeling, by Max Kuhn and Kjell Johnson
- Machine Learning Mastery with Python, by Jason Brownlee
- Deep Learning with Python, by Jason Brownlee
- The Elements of Statistical Learning, by Trevor Hastie, Robert Tibshirani, and Jerome Friedman

Also, the online documentation of the Python libraries used in this course is a great place to learn more. Here are some of the most relevant libraries that are used throughout the course:

- matplotlib
- numpy
- pandas
- seaborn
- scikit-learn
- statsmodels

Teaching Assistants. A group of TAs have agreed to help with this class. The TAs are available to help students and will have office hours. Please feel free to contact the TAs directly. Their contact information is available on Canvas.



Schedule.

Module	Topic	Submodules
1 Week of January 20	Regression models	1. Beyond simple linear regression: multiple linear regression 2. Logistic regression 3. Generalized linear models
2 Week of February 3	Regularization and model tuning	1. Overfitting 2. Regularization: LASSO and Ridge regression 3. Data splitting and cross-validation
3 Week of February 17	Classification	1. Introduction to classification and decision boundaries 2. Naïve Bayes and k-nearest neighbors 3. Support vector machines
4 Week of March 2	Tree-based methods and ensembles	1. Classification trees 2. Bagging and random forests 3. Boosting
5 Week of March 16	Unsupervised learning	1. Introduction and k-means 2. Hierarchical clustering 3. Density-based clustering 4. Dimensionality reduction
6 Week of March 30	Introduction to deep learning Data analytics projects in the real world Course wrap-up and recap	1. Introduction to Deep Learning 2. Data analytics projects in the real world 3. Course wrap-up and recap

Questions?

Introduction to Predictive Modeling:
Linear Regression (and Generalizations)

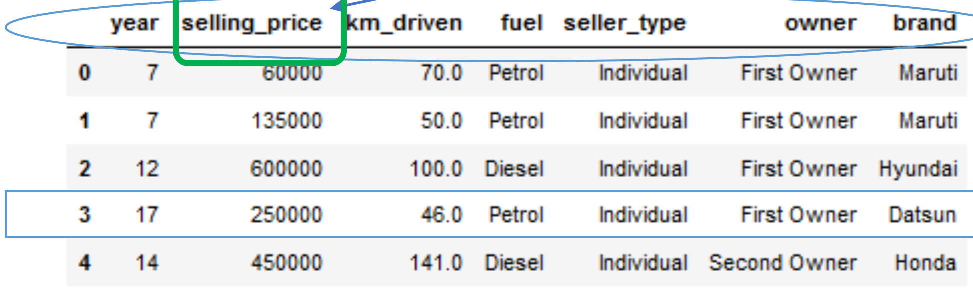
“Supervised” vs. “Unsupervised” problems

- Supervised problems involve a specific *target* (or response) variable, and the goal is to understand the relationship of the target to other feature variables
  - Examples:
    - “Can we find groups of bank customers that have a higher probability of subscribing to a term deposit?”
    - “Based on digital image data, can we tell which patients have a higher probability of having a malignant tumor?”
  - Most supervised learning problems involve *predictive modeling* techniques, such as classification and regression.

- Unsupervised problems do not have a target variable, and the goal is to understand the structure of the data.
  - Examples:
    - “Do there exist natural groups of bank customers? If so, what features distinguish these groups?”
    - “Do individuals with a particular disease fall into natural groups of risk factors?”
  - *Clustering* is a canonical unsupervised learning problem.

Predictive modeling: finding a model of a target variable in terms of descriptive attributes

- Example:



	year	selling_price	km_driven	fuel	seller_type	owner	brand
0	7	60000	70.0	Petrol	Individual	First Owner	Maruti
1	7	135000	50.0	Petrol	Individual	First Owner	Maruti
2	12	600000	100.0	Diesel	Individual	First Owner	Hyundai
3	17	250000	46.0	Petrol	Individual	First Owner	Datsun
4	14	450000	141.0	Diesel	Individual	Second Owner	Honda

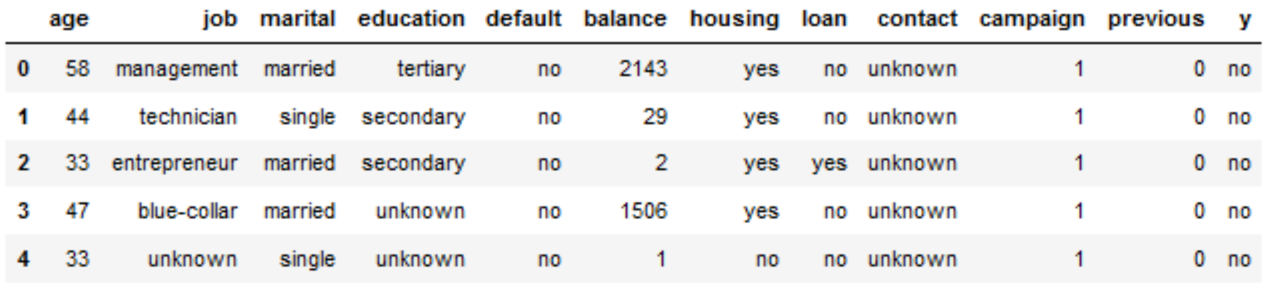
Target attribute

Attributes

Each row is an example. For this example:

- Feature vector = (17, 46.0, Petrol, Individual, First Owner, Datsun)
- Target value = 250,000

- Another example:

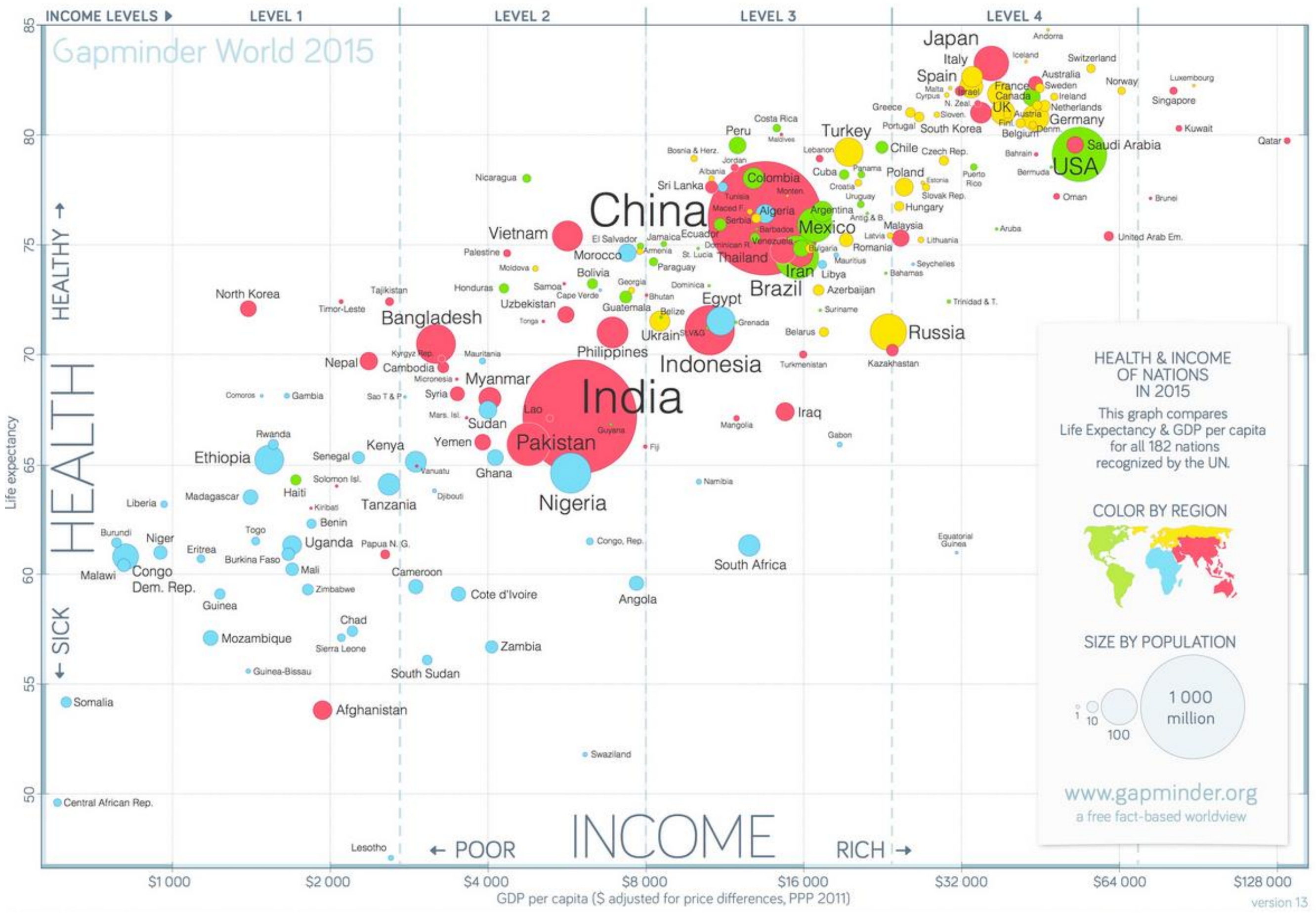


	age	job	marital	education	default	balance	housing	loan	contact	campaign	previous	y
0	58	management	married	tertiary	no	2143	yes	no	unknown	1	0	no
1	44	technician	single	secondary	no	29	yes	no	unknown	1	0	no
2	33	entrepreneur	married	secondary	no	2	yes	yes	unknown	1	0	no
3	47	blue-collar	married	unknown	no	1506	yes	no	unknown	1	0	no
4	33	unknown	single	unknown	no	1	no	no	unknown	1	0	no

- There are many kinds of predictive models, and broadly they fall into two categories:
  - Parametric: assume a model structure with some unspecified numbers; use data to fit the numbers.
  - Non-parametric: model structure determined directly from data.

Regression is used in many applications to understand relationships between various quantities

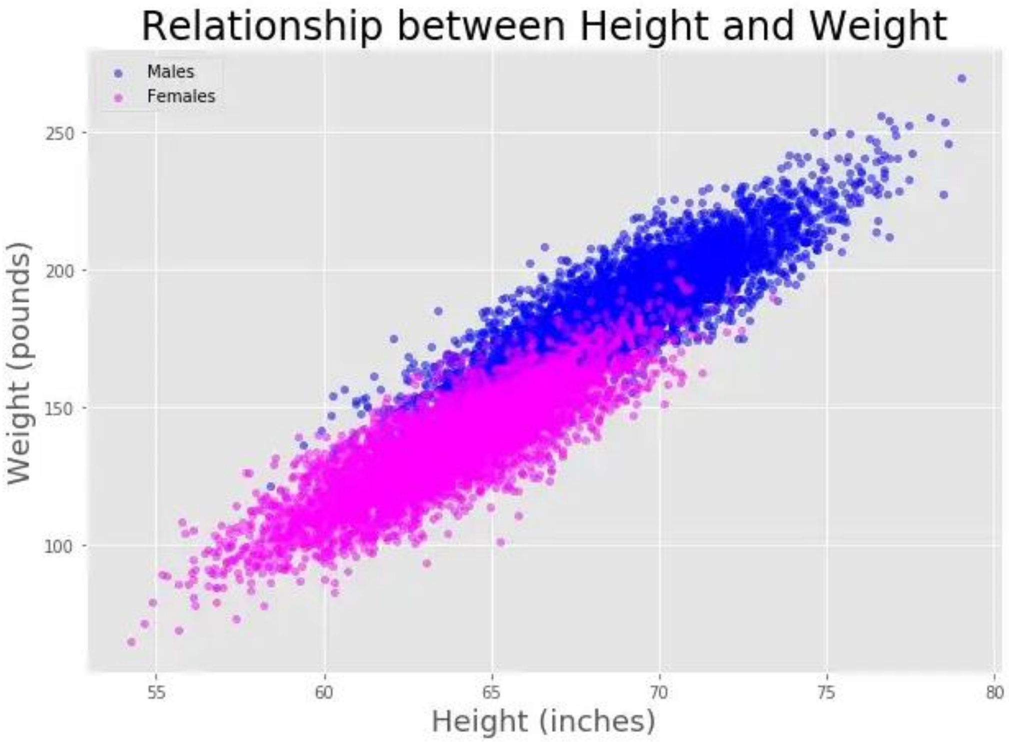
- Relationship between GDP per capita and life expectancy:



Source: *https://www.gapminder.org/tag/life-expectancy/*

Another example

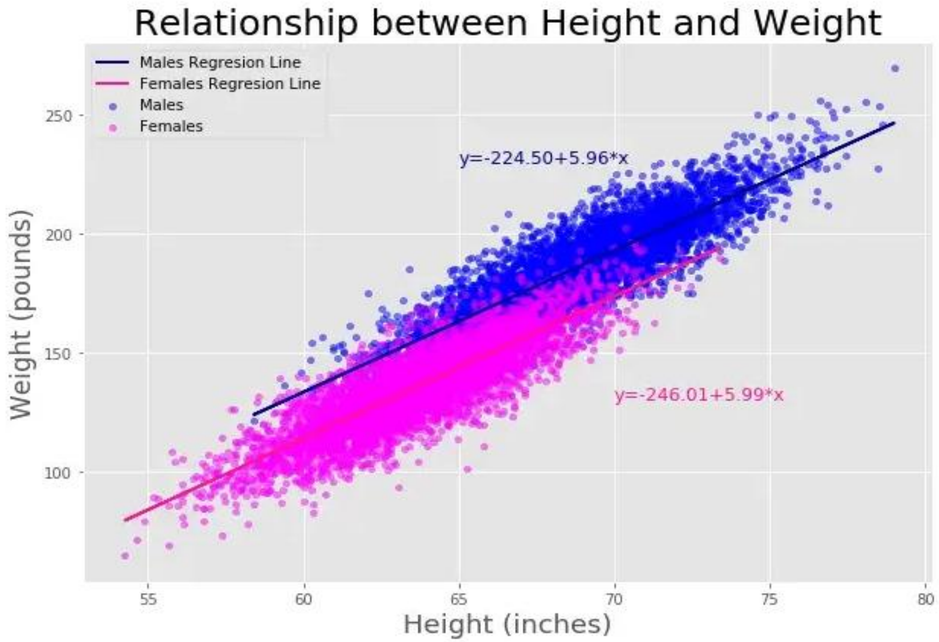
- Example: the relationship between a person's height and weight



Source: https://towardsdatascience.com/simple-and-multiple-linear-regression-with-python-c9ab422ec29c

Example (continued)

Seaborn



Source: https://towardsdatascience.com/simple-and-multiple-linear-regression-with-python-c9ab422ec29c

- Is there a (linear) relationship?
- Various quantities related to statistical inference: p-values, t-stats, measures of fit (R-squared, mean squared error), etc.
- In data analytics, we care more about how well the model *predicts* new and unseen target values on the basis of observed features

Linear regression

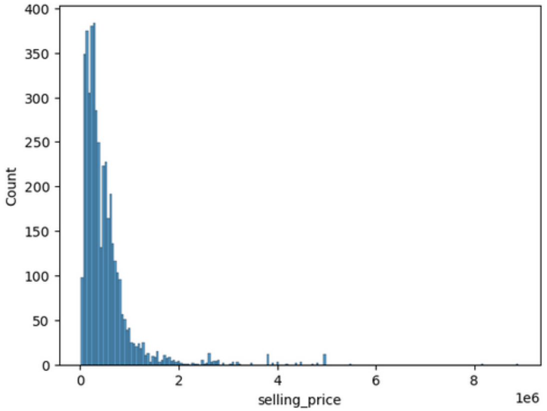
- We are trying to predict the value of a *target variable* (call it Y) given information about *feature variables* (call them $X_1, X_2, \ldots$)
- The regression equation:

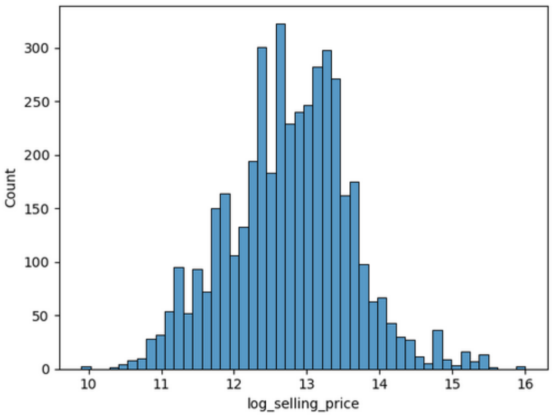
$$Y = w_0 + w_1 * X_1 + w_2 * X_2 + \cdots \ (+ Error)$$

- Assumptions?
- The weights (w_i) are design parameters: the regression algorithm chooses these to minimize the sum of squared errors on given data $\left(Y_i, X_{1,i}, X_{2,i}, \ldots\right)$
- Why the sum of *squared* errors?

In many problems, we need to transform the data before fitting a regression model

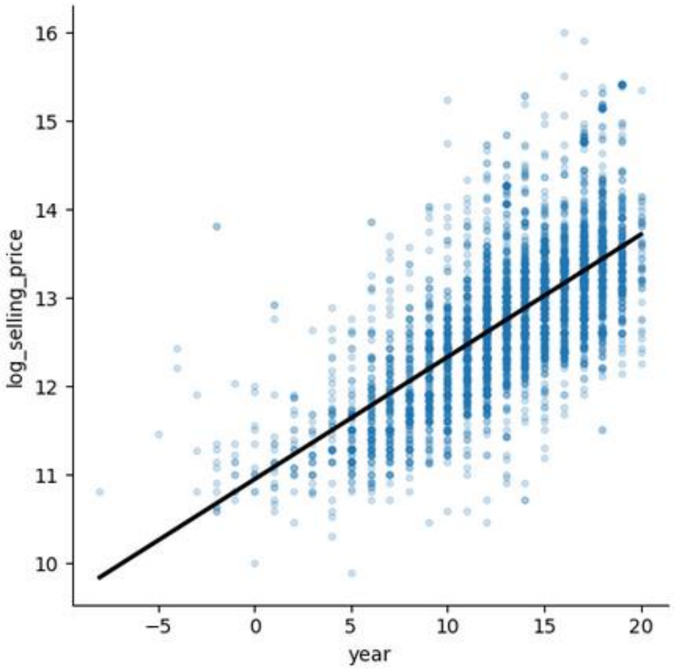
- Example transformations:
    - Feature variables: turning categorical features into binary (“dummy”) variables, scaling or shifting numerical features, taking products of binary and numerical variables (these are called interaction effects)
    - Target variable: various nonlinear transformations
- Many common target variables are nonnegative (e.g., prices, revenues, sales, life expectancy) and have a “skewed” distribution. A log transformation is common in these settings:

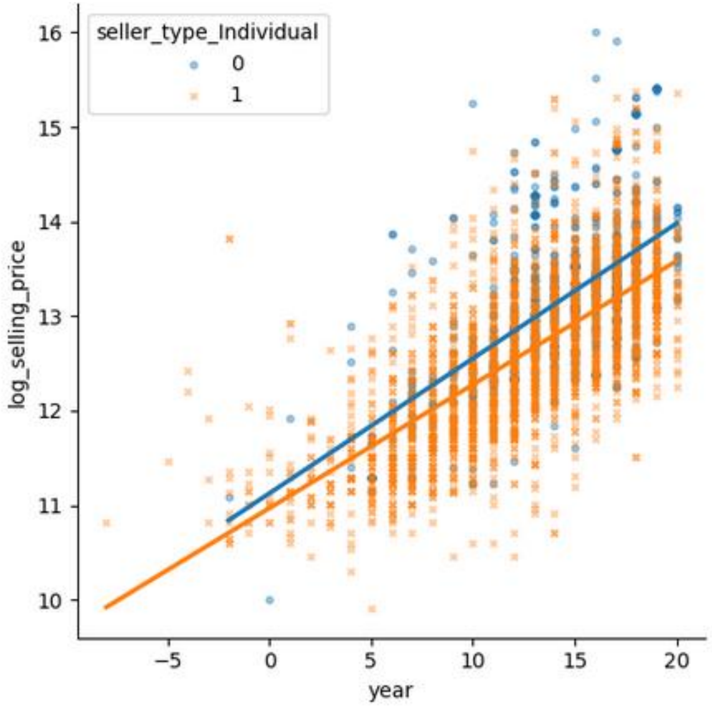




Assessing the quality of a regression model

- CarDekho example:





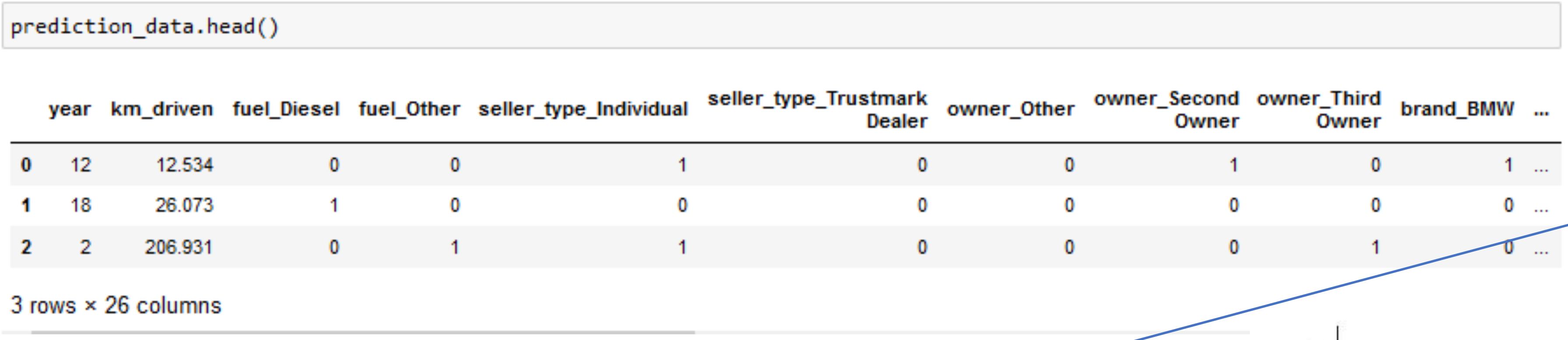
Simple linear regression; R-squared ~ 0.48

Multiple linear regression; R-squared ~ 0.51

- Which is model is better at predicting future prices?

Our focus with regression models will be to make predictions

- More on the CarDekho example:



```
prediction_data.head()
```

	year	km_driven	fuel_Diesel	fuel_Other	seller_type_Individual	seller_type_Trustmark Dealer	owner_Other	owner_Second Owner	owner_Third Owner	brand_BMW	...
0	12	12.534	0	0	1	0	0	1	0	1	...
1	18	26.073	1	0	0	0	0	0	0	0	...
2	2	206.931	0	1	1	0	0	0	1	0	...

3 rows × 26 columns

New feature vectors

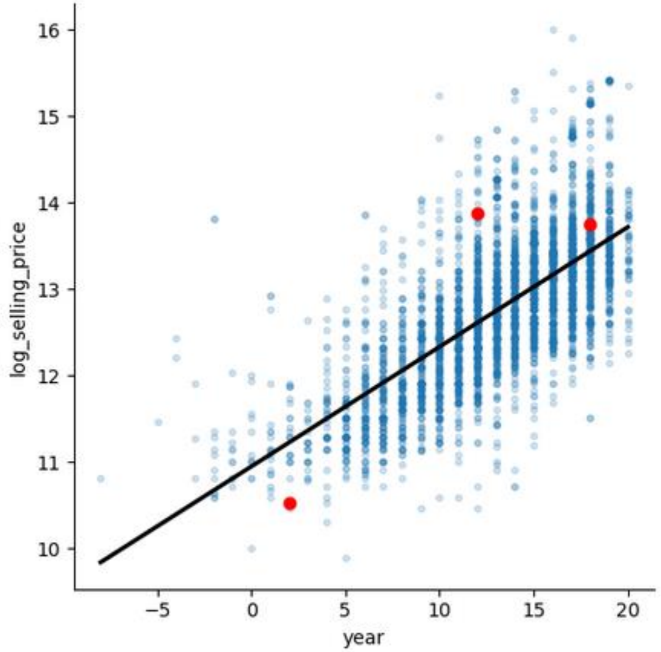
Predicted log selling prices

13.88

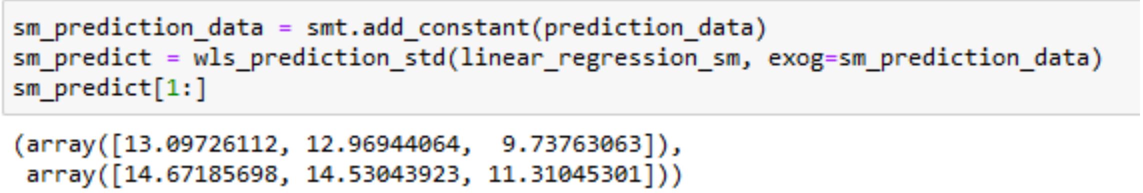
13.75

10.52

- *What do these predictions mean*?



We can also get *prediction intervals*.
Example intervals at 95%:
(need from statsmodels.sandbox.regression.predstd import wls_prediction_std)

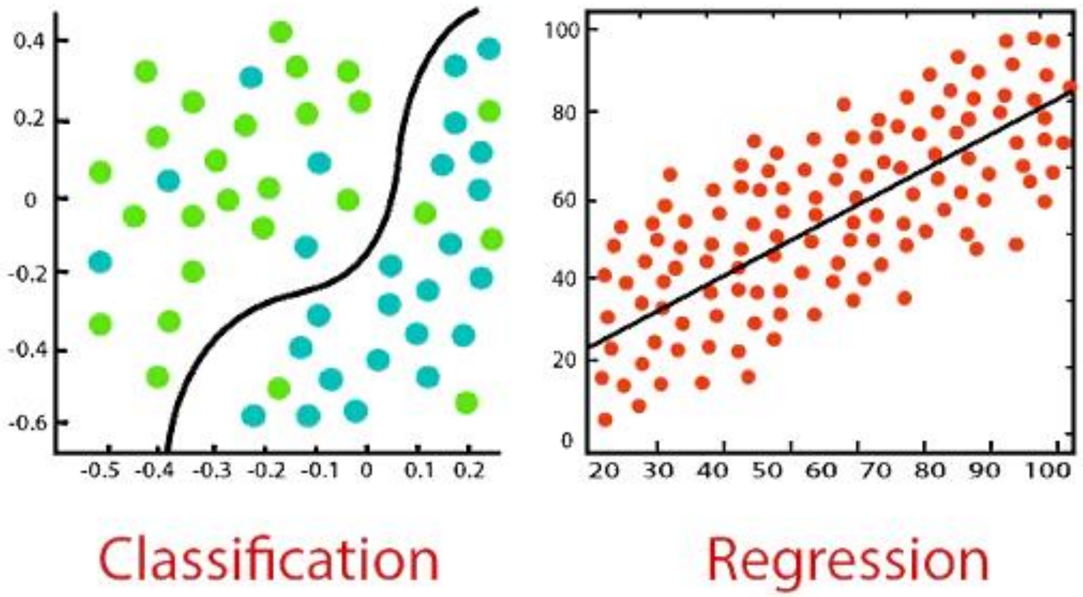


```
sm_prediction_data = smt.add_constant(prediction_data)
sm_predict = wls_prediction_std(linear_regression_sm, exog=sm_prediction_data)
sm_predict[1:]
```

```
(array([13.09726112, 12.96944064,  9.73763063]),
 array([14.67185698, 14.53043923, 11.31045301]))
```

In many settings, we want to estimate the probability of something happening

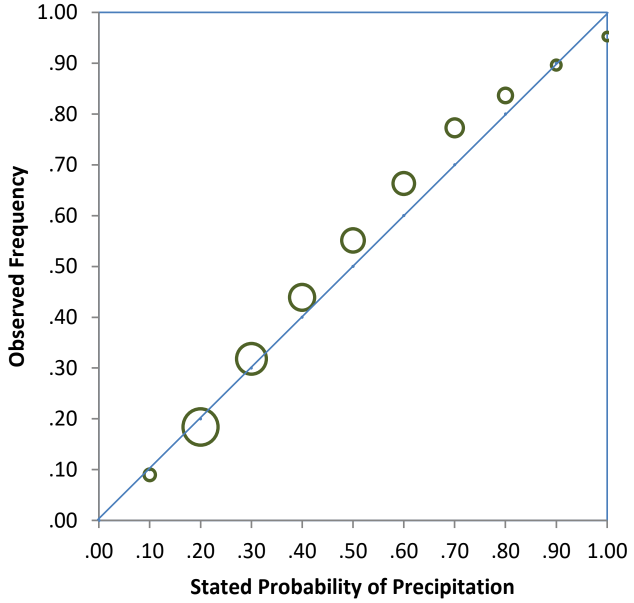
- This relates to classification (or *class probability estimation*).



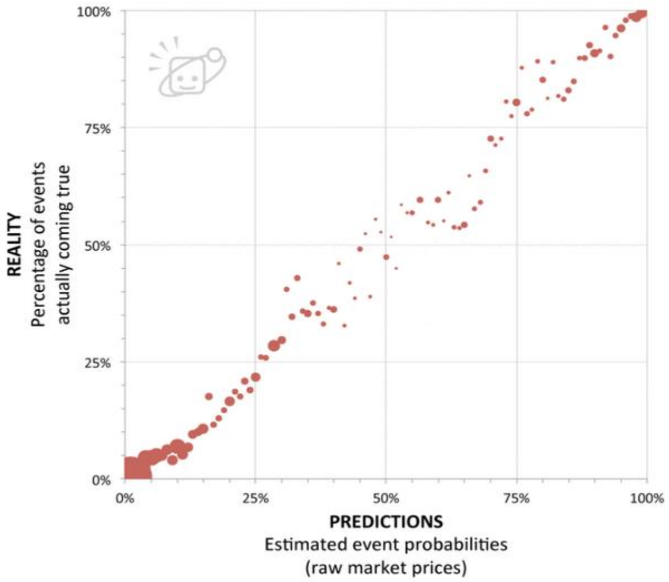
- What does it mean for a probability estimate to be “good”?

Goodness of probability estimates

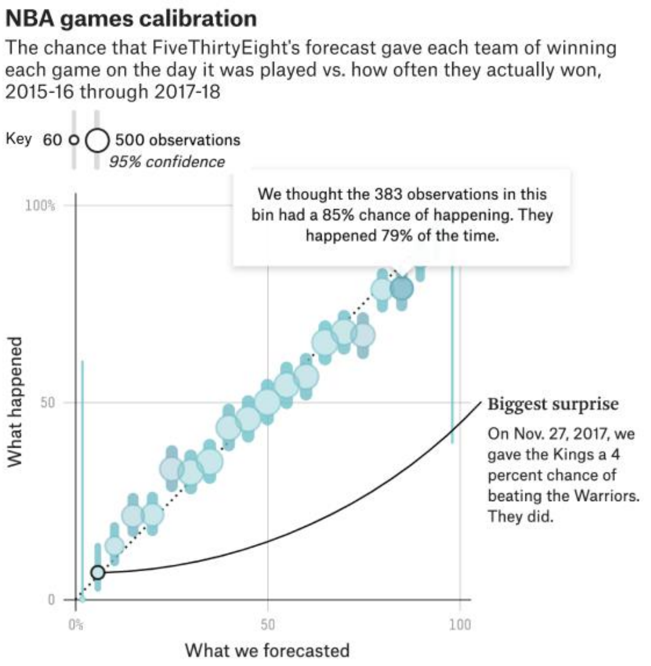
- We would like probability estimates to be:
  - Calibrated: of the times we say 30%, the outcome occurs 30% of the time
  - Informative: estimates close to 0 and 1



A calibration plot for U.S. National Weather Service Forecasters for day ahead Probability of Precipitation Forecasts (248,348 observations). Circle sizes are proportional to the frequency of the stated forecasts. Note: Typically, no forecast is issued if the probabilities are low; thus there is no data for 0% and little for 10% probabilities. Source: Bickel, Floehr, Kim (2011)



Calibration plot from Hypermind (a prediction market) for 213 “electoral, geopolitical, and economic questions” from 2014-2016. Size of points reflects number of forecasts (56,949 total forecasts).


NBA games calibration

The chance that FiveThirtyEight's forecast gave each team of winning each game on the day it was played vs. how often they actually won, 2015-16 through 2017-18

We thought the 383 observations in this bin had a 85% chance of happening. They happened 79% of the time.

Biggest surprise

On Nov. 27, 2017, we gave the Kings a 4 percent chance of beating the Warriors. They did.

Source: “When We Say 70 Percent, It Really Means 70 Percent” by Nate Silver.

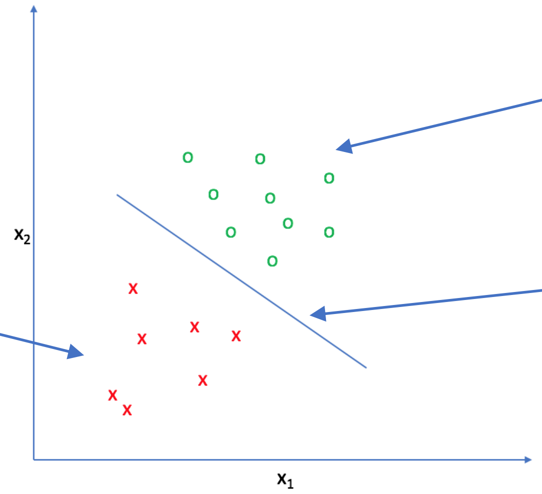
https://fivethirtyeight.com/features/when-we-say-70-percent-it-really-means-70-percent/

Logistic regression is a canonical technique for class probability estimation

- Marketing example: “What is the probability a customer will accept a promotional offer given their age, account balance, ZIP code, etc.?”

Past customers who accepted offer

Past customers who rejected offer



Regression line:
$g(x) = w_0 + w_1x_1 + w_2x_2 + \ldots$

Here $g(x)$ is a “score” telling us whether accept or reject is more likely depending on customer features x.

- We can use linear regression as a starting point:
  - Points farther from regression line are more/less likely to accept offer. Points exactly on line are 50-50 cases.
  - Problem: the value $g(x)$ need not be between 0 and 1 (e.g., could be negative).

We can “fix” this by considering a different target variable

- Want: probability that customer with features x accepts; call this $f(x)$
  1. The odds ratio (or just “odds”) is then $f(x)/(1-f(x))$, which is between 0 and ∞
  2. The log of the odds ratio (just log-odds) is then between $-\infty$ and ∞

  Example:

Probability	**Odds**	**Log-odds**
0.5	**1:1 (= 1)**	**0**
0.9	**9:1 (= 9)**	**2.19**
0.999	**999:1 (= 999)**	**6.90**
0.01	**1:99 (= 0.0101)**	**-4.6**
0.001	**1:999 (= 0.001001)**	**-6.9**

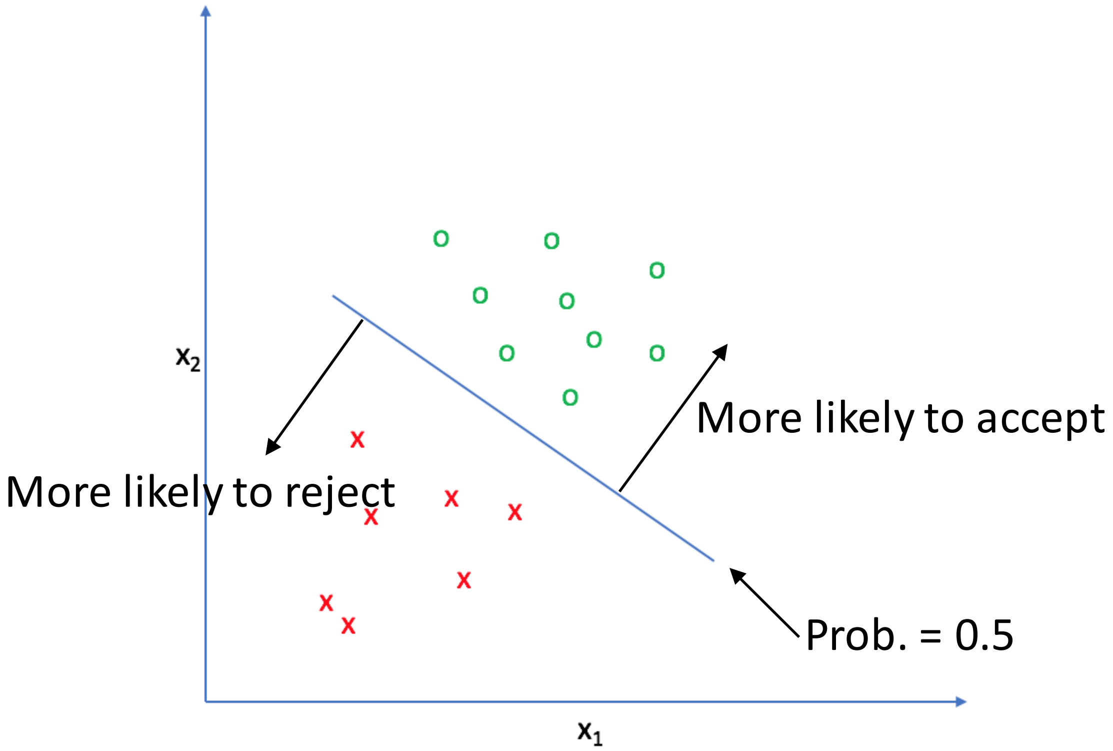
“Logit function” (hence the name *logistic* regression)

- Logistic regression equation: $\log\frac{f(x)}{1-f(x)} = g(x) = w_0 + w_1x_1 + w_2x_2$

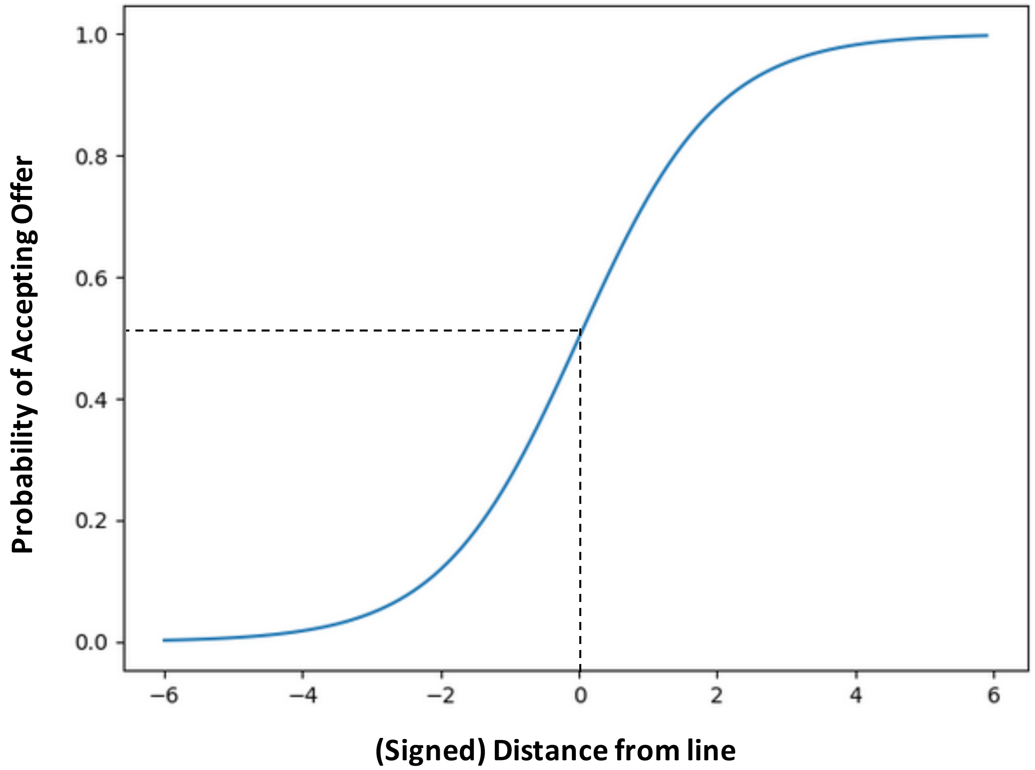
  which is the same as: $f(x) = \frac{1}{1+e^{-g(x)}} = \frac{1}{1+e^{-(w_0+w_1x_1+w_2x_2)}}$

Visualizing a logistic regression model

View of data:



View of fitted logistic model:



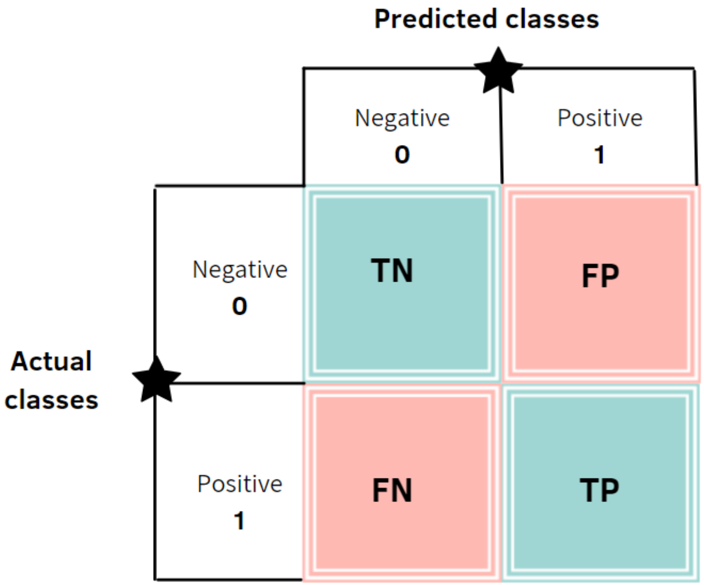
- Customers "on the line" (distance = 0): 0.5 probability of accepting
- Customers farther from line are more or less likely to accept

In classification problems, we want to predict whether a certain outcome will happen or not

- Example: “Will customer with features x accept this promotional offer or not?”
- Key difference from regression: target variable is categorical: “yes” (1) or “no” (0)
- We can use logistic regression for classification. Rule:
  1. Given features, if estimated probability is above a threshold, predict the target will be “yes.”
  2. Otherwise, predict the target will be “no.”
- A natural threshold is 50%, but this may not be the best choice; in practice, we can vary the threshold for the problem at hand.

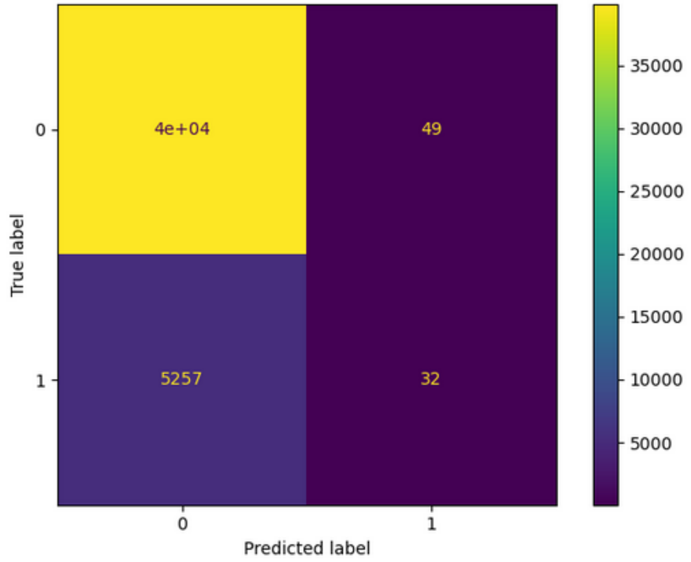
Confusion matrices summarize classification performance

Picture of confusion matrix:



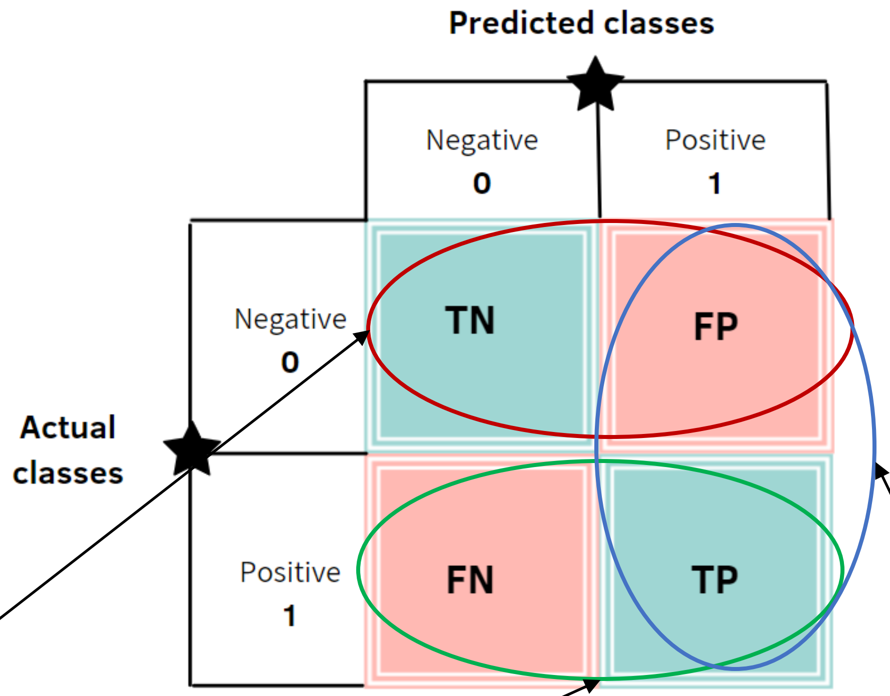
		Predicted classes	
		Negative 0	Positive 1
Actual classes	Negative 0	TN	FP
	Positive 1	FN	TP

From `sklearn` (bank data example):



- *Warning:* some people flip axes (true = horizontal, predicted = vertical)
- Classifier Accuracy = $\dfrac{\text{TN+TP}}{\text{TN+FN+TP+FP}}$ = (correct classifications)/(all classifications)

Common classification metrics (many others!)



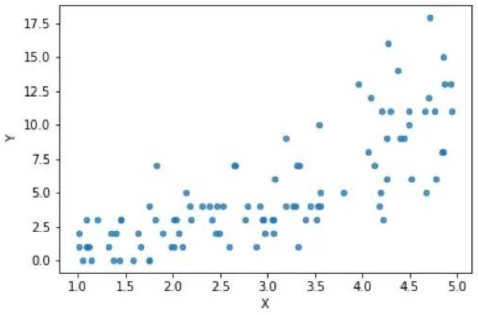
Sensitivity = Recall = $\frac{\text{TP}}{\text{FN+TP}}$ = “true positive rate”

Specificity = $\frac{\text{TN}}{\text{TN+FP}}$ = “true negative rate”

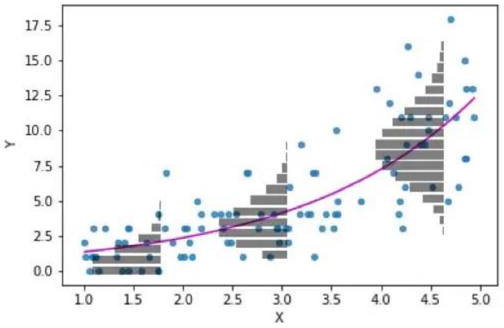
Precision = $\frac{\text{TP}}{\text{FP+TP}}$ = “positive predictive value”

We can “generalize” linear regression in many ways

- Would you use linear regression here?



- Generalized linear models (GLM):
  1. How does the target Y change on average with changes in features X? (“Inverse link function”)
  2. What restrictions are there on the values of Y (i.e., nonnegative, integer, binary)? What probability distribution should Y follow?
- Poisson regression on the above example:
- Many choices are possible. Use:
  - Data analytics expertise
  - Knowledge of problem domain
  - Trial and error!



Source: https://towardsdatascience.com/generalized-linear-models-9cbf848bb8ab

GLM recipe

“Inverse link function”

- Components of a GLM model:

1. $$E[Y|X] = f(w_0 + w_1 * X_1 + w_2 * X_2 + \cdots)$$

“Link function” (f and g are inverses)

This is the same as:

$$g(E[Y|X]) = w_0 + w_1 * X_1 + w_2 * X_2 + \cdots$$

2. Given features X, the target Y follows a particular probability distribution.

- Common examples:

Name	Link $g(y)$	Inverse link $f(x)$	Distribution
Linear	Identity ($= y$)	Identity ($= x$)	Normal
Logistic	Logit ($= \ln\left(\frac{y}{1-y}\right)$)	“S-curve” $\frac{1}{1+e^{-x}}$	Bernoulli (or “yes-no”)
Gamma	$\frac{1}{y}$	$\frac{1}{x}$	Gamma
Poisson	$\ln(y)$	e^x	Poisson

Interpretations of GLM coefficients

- *Always true*:

✓ $$\boxed{g(E[Y|X_i = x_i + 1]) - g(E[Y|X_i = x_i]) = w_i}$$

- With a bit of algebra, we can often simplify the above interpretation. Examples:
  - additive Linear regression: "When X_i increases by one unit, the target Y increases by w_i on average, all else equal."
  - multiplicative Logistic regression: "When X_i increases by one unit, the odds ratio of the target event Y occurring is multiplied by e^{w_i}, all else equal."

Summary

- Most supervised learning problems involve building a predictive model: we want to predict the value of some unseen target variable on the basis of observed feature variables.
- We have discussed parametric regression models: assume a model structure, then fit numbers using data.
- Linear regression is a widely used form of predictive modeling.
- Logistic regression is a form of linear regression used to predict the probability that a binary target variable is a “yes” or a “no.”
    - Logistic regression is also a form of classification; confusion matrices provide a summary of classifier performance.
- Generalized linear models (GLM) allow us flexibility to go beyond the usual linear regression framework.

Looking ahead to next time (Class 2)

- Homework 1 due at 11:59pm on Monday
  - Main goals: test your understanding of regression and classification models.
  - TA support available over the weekend!

- Class 2:
  - How do we select feature variables? Are more variables better?
  - How do we select a “best” predictive model from a group?
  - How do we evaluate a predictive model’s performance before deploying it?