BTS | Barcelona Technology School

# Data Science Foundations


**Master in Big Data Solutions 2020-2021**

Víctor Pajuelo

victor.pajuelo@bts.tech

# Today's class

# Contents

- End 2 “End” machine learning project

# Today's objective

- To be able to go from data mess to data insights
- Keep the ideas of data science driven business clear

# End 2 “End” Machine Learning project

# E2E ML Project

## Getting hired as a Data Scientist

- We will pretend that you got hired as a data scientist
- We will use some housing dataset in order to pretend that you got hired in a disrupting housing company called…
  - The ML HOUSING CORPORATION

- But first, a project checklist that you need to have in mind before any project

- And welcome to the ML HOUSING CORPORATION!

# E2E ML Project

## ML project checklist

This checklist can guide you through your Machine Learning projects. There are eight main steps:

1. Frame the problem and look at the big picture.
2. Get the data.
3. Explore the data to gain insights.
4. Prepare the data to better expose the underlying data patterns to Machine Learning algorithms.
5. Explore many different models and short-list the best ones.
6. Fine-tune your models and combine them into a great solution.
7. Present your solution.
8. Launch, monitor, and maintain your system.

# End 2 “End” ML Project Checklist

# Frame the problem and look at the big picture

# E2E ML Project

## 1. Frame the Problem and Look at the Big Picture

1. Define the objective in business terms.
2. How will your solution be used?
3. What are the current solutions/workarounds (if any)?
4. How should you frame this problem (supervised/unsupervised, online/offline, etc.)?
5. How should performance be measured?
6. Is the performance measure aligned with the business objective?
7. What would be the minimum performance needed to reach the business objective?
8. What are comparable problems? Can you reuse experience or tools?
9. Is human expertise available?
10. How would you solve the problem manually?
11. List the assumptions you (or others) have made so far.
12. Verify assumptions if possible.

# E2E ML Project

## Looking at the Big Picture

- If you have used already preprocessed datasets before, don't you worry!
  - You can either use the datasets from your colleagues, or download others. We will do a trial run with California datasets first.
- The first task that you will be asked in this company is to build a model of housing prices using census data
- The dataset that we have has metrics such as the population, median income, median housing price, and so on for each block group in California. Block groups are the smallest geographical unit for which the US Census Bureau publishes sample data (a block group typically has a population of 600 to 3,000 people). We will just call them "districts" for short.
- Your model should learn from this data and be able to predict the median housing price in any district, given all the other metrics.

# E2E ML Project

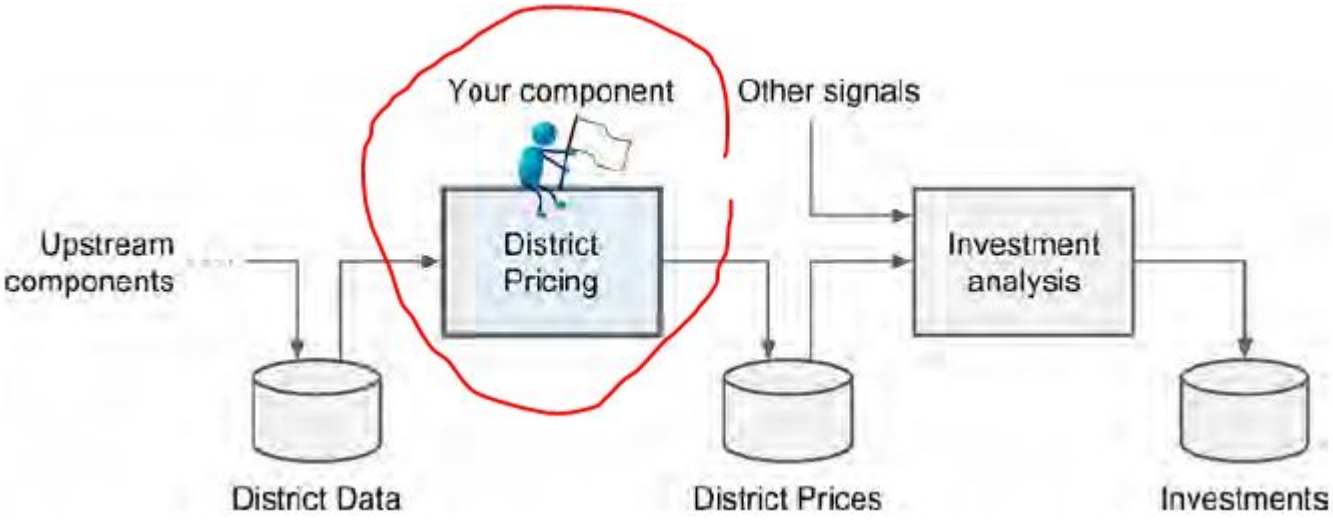
## Frame the Problem

- The first question to ask your boss is **what exactly is the business objective**; building a model is probably not the end goal. How does the company expect to use and benefit from this model?
  - This will determine how you frame the problem, what algorithms you select, what performance measure you will use for evaluation, how much effort you should spend tweaking the model
- Your boss answers you that **your model's output will be fed to another Machine Learning model**, so you will need to build a **pipeline**

# E2E ML Project

## The ML pipeline

- A sequence of data processing *components* is called a data *pipeline*. Pipelines are very common in Machine Learning systems, since there is a lot of data to manipulate and many data transformations to apply



# E2E ML Project

## Select a Performance Measure

- You will need to select performance measures that are fit to your problem
  - In terms of regression Root Means Square Error or Mean Absolute Error will be some of the ones that you could choose
  - In terms of classification Precision/Recall, ROC curves, accuracy

- For a complete list of the performance metrics that you can use with Sklearn, see here: https://scikit-learn.org/stable/modules/model_evaluation.html

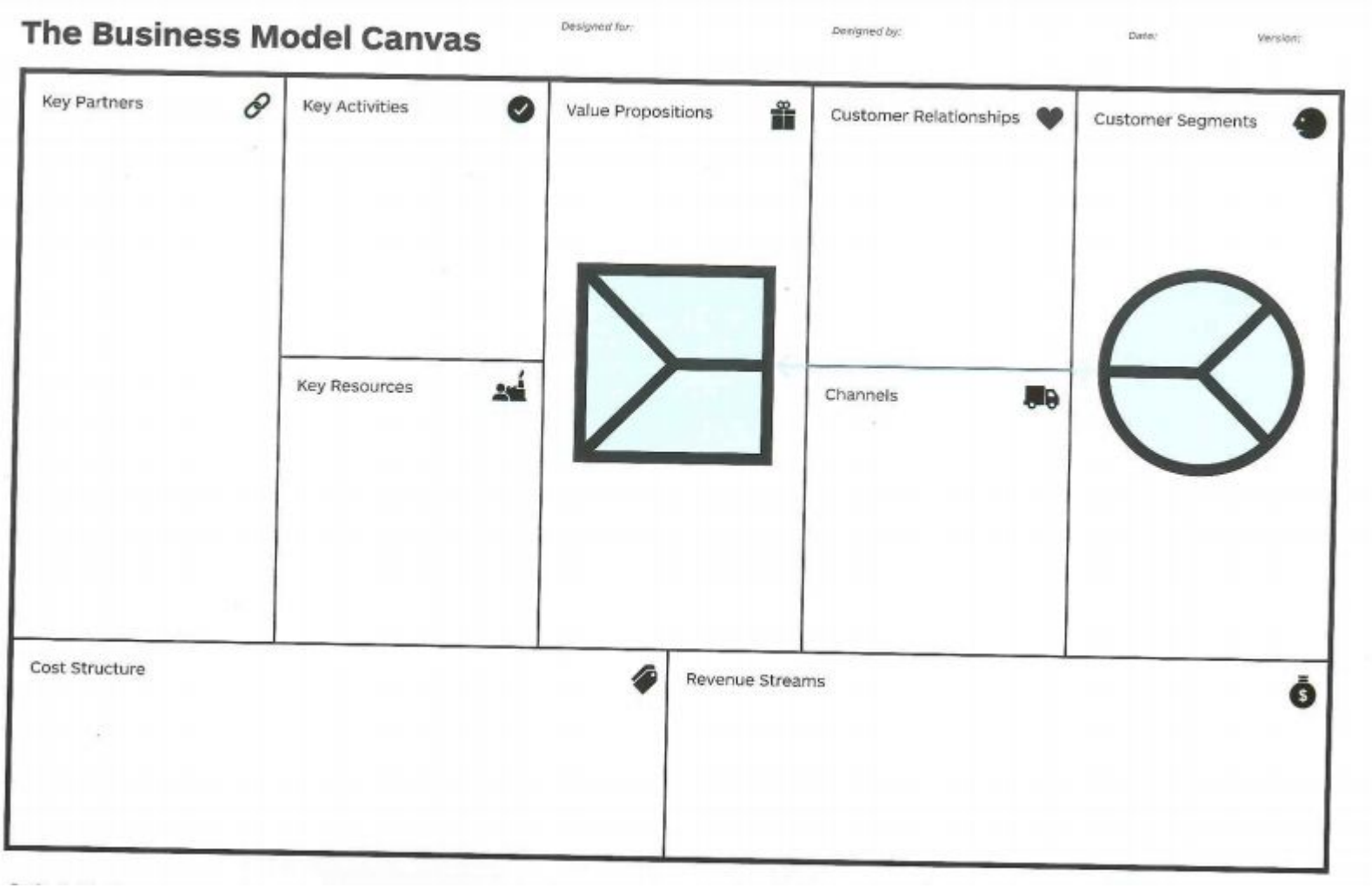
# E2E ML Project

## Check the Assumptions

- Verify your assumptions with your manager, to catch serious issues earlier on
- The district prices that our algorithm will produce will be fed onto another machine learning pipeline
  - If we assume that the district prices will be a continuous variable, but we do not double check it, we might end up with a problem
  - What if instead a regression task, we need to do a classification? Maybe instead of a continuous variable (price) we need a category such as *cheap, medium* or *expensive*
- Always double check, you will not believe how many times this issues propagate through the development line

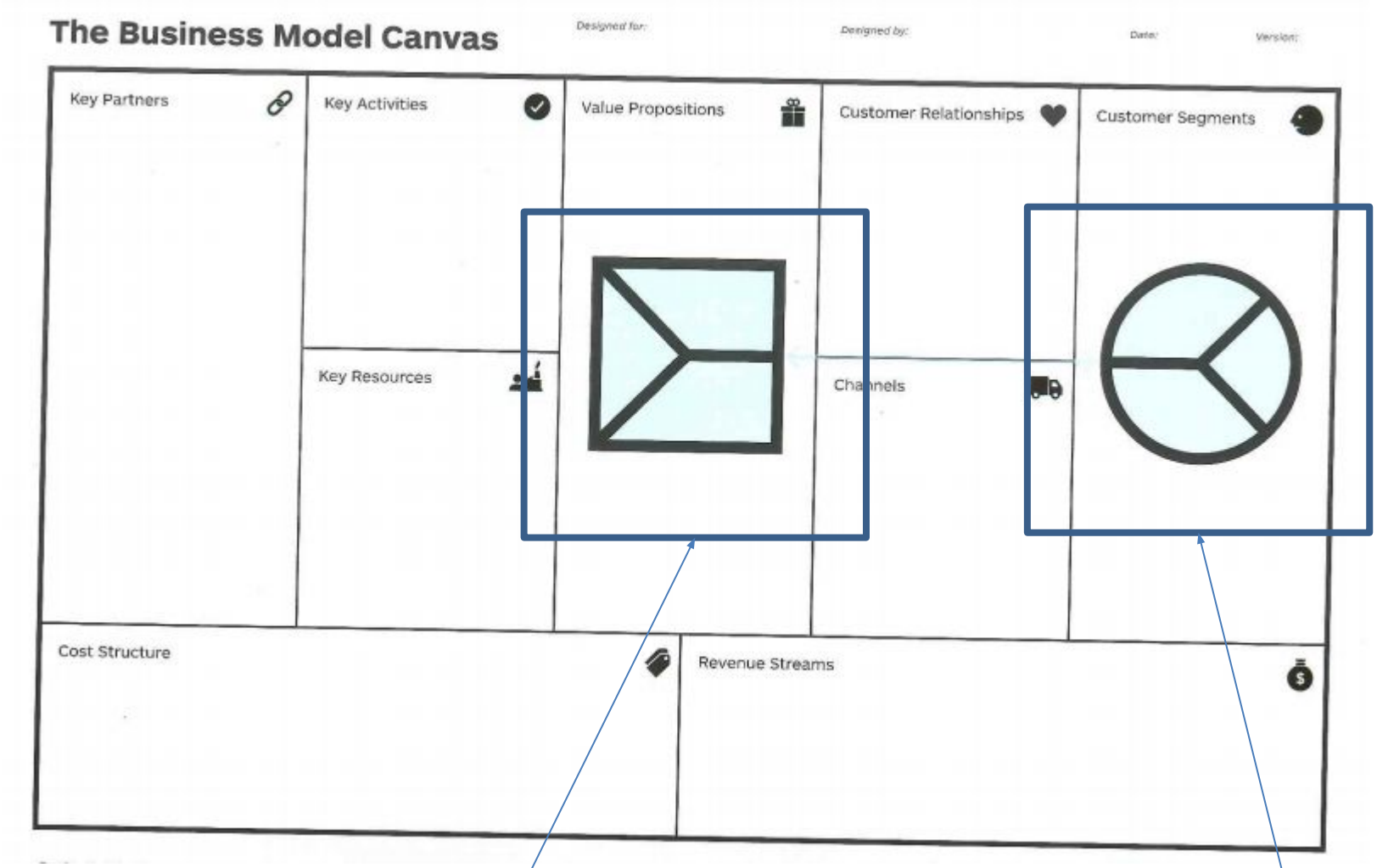
# E2E ML Project

## A valuable Data Scientist is a Business Developer with superpowers



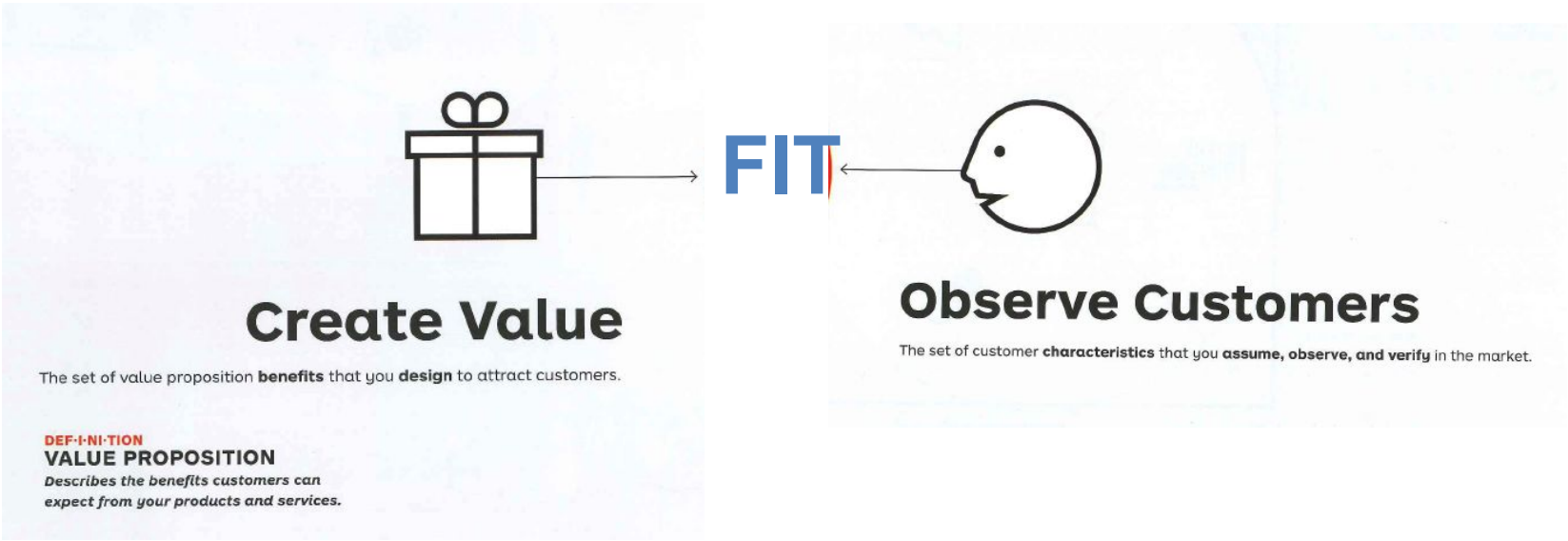
# E2E ML Project

## A valuable Data Scientist is a Business Developer with superpowers



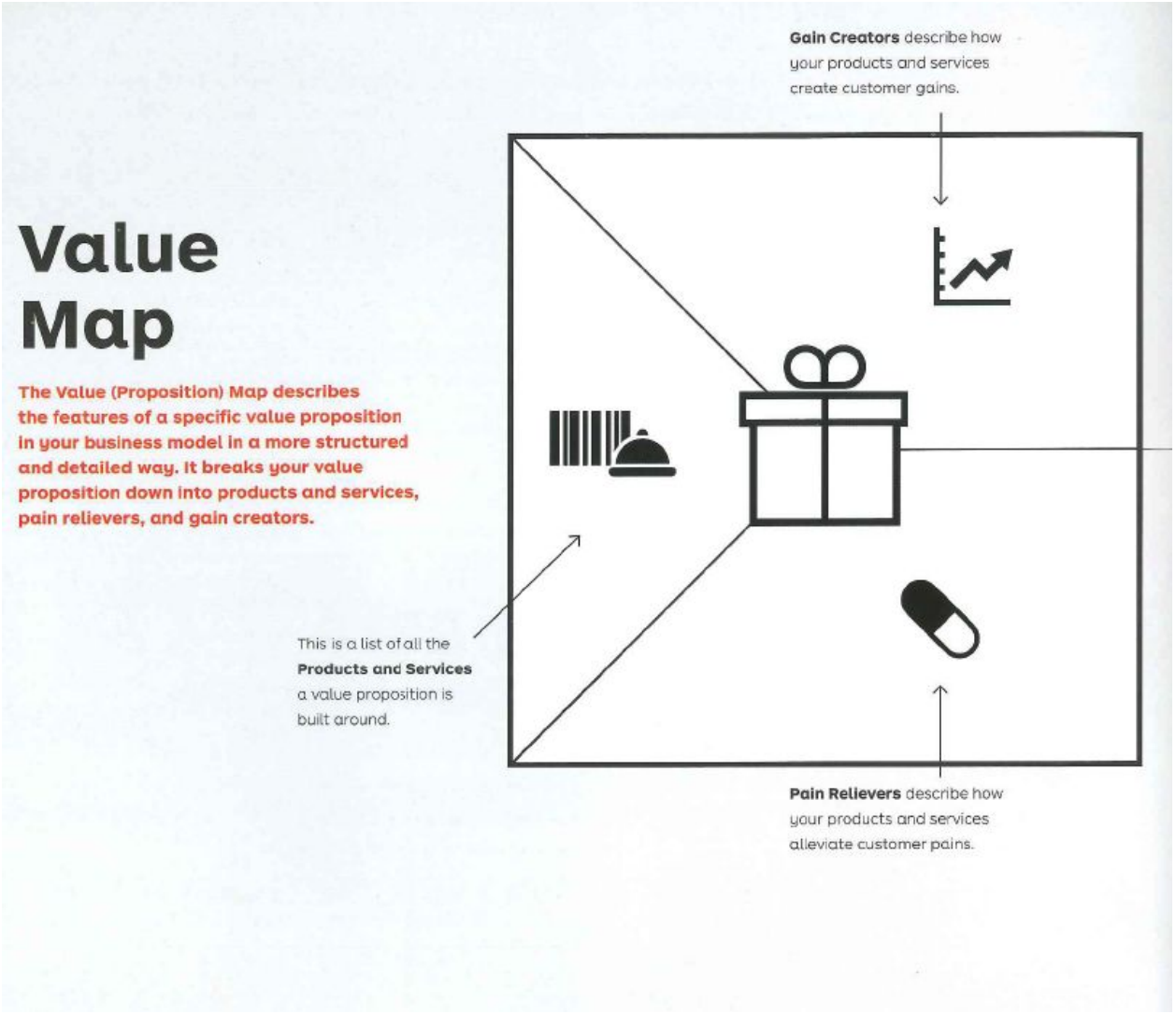
# E2E ML Project

## A valuable Data Scientist is a Business Developer with superpowers



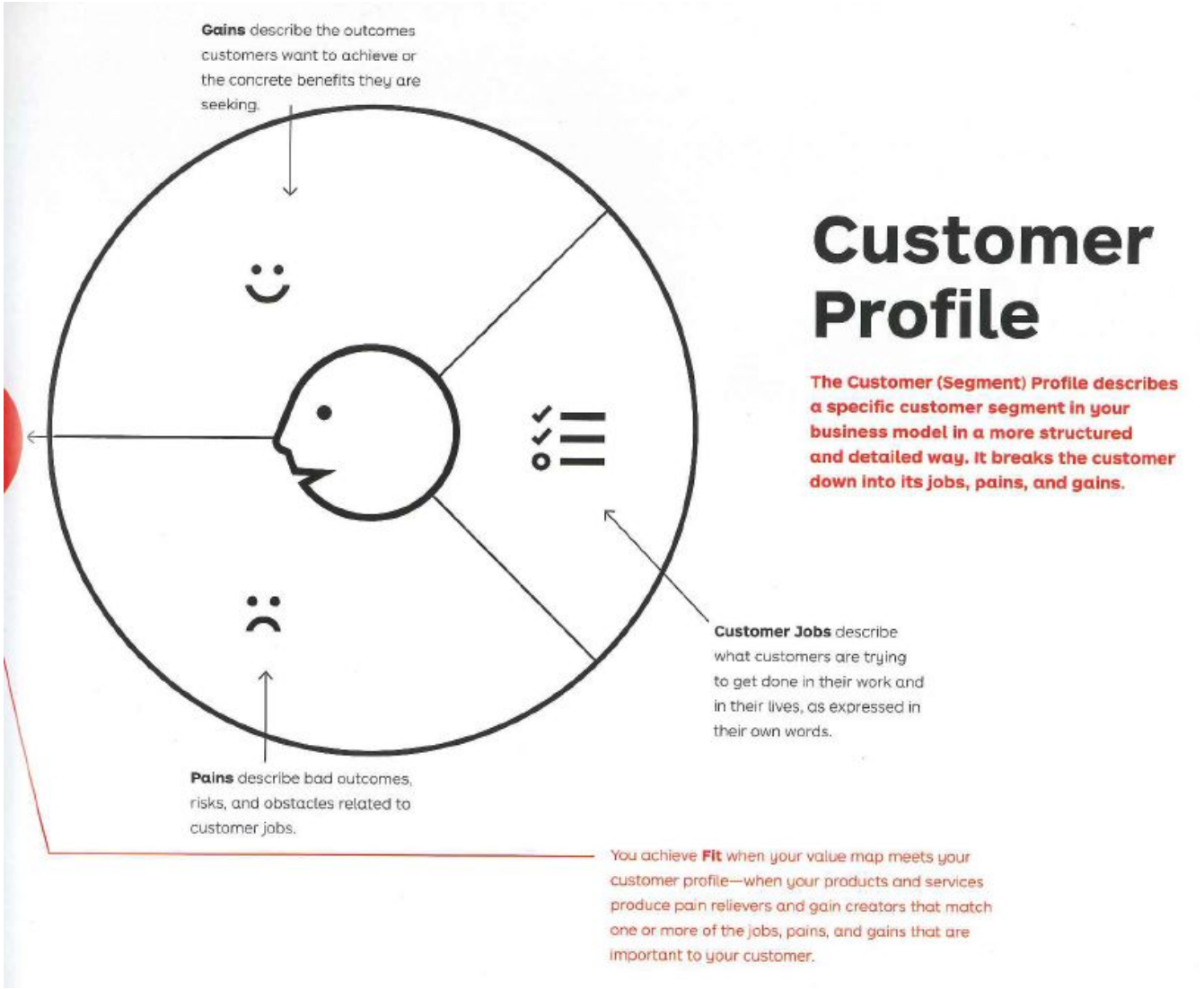
# E2E ML Project

## A valuable Data Scientist is a Business Developer with superpowers



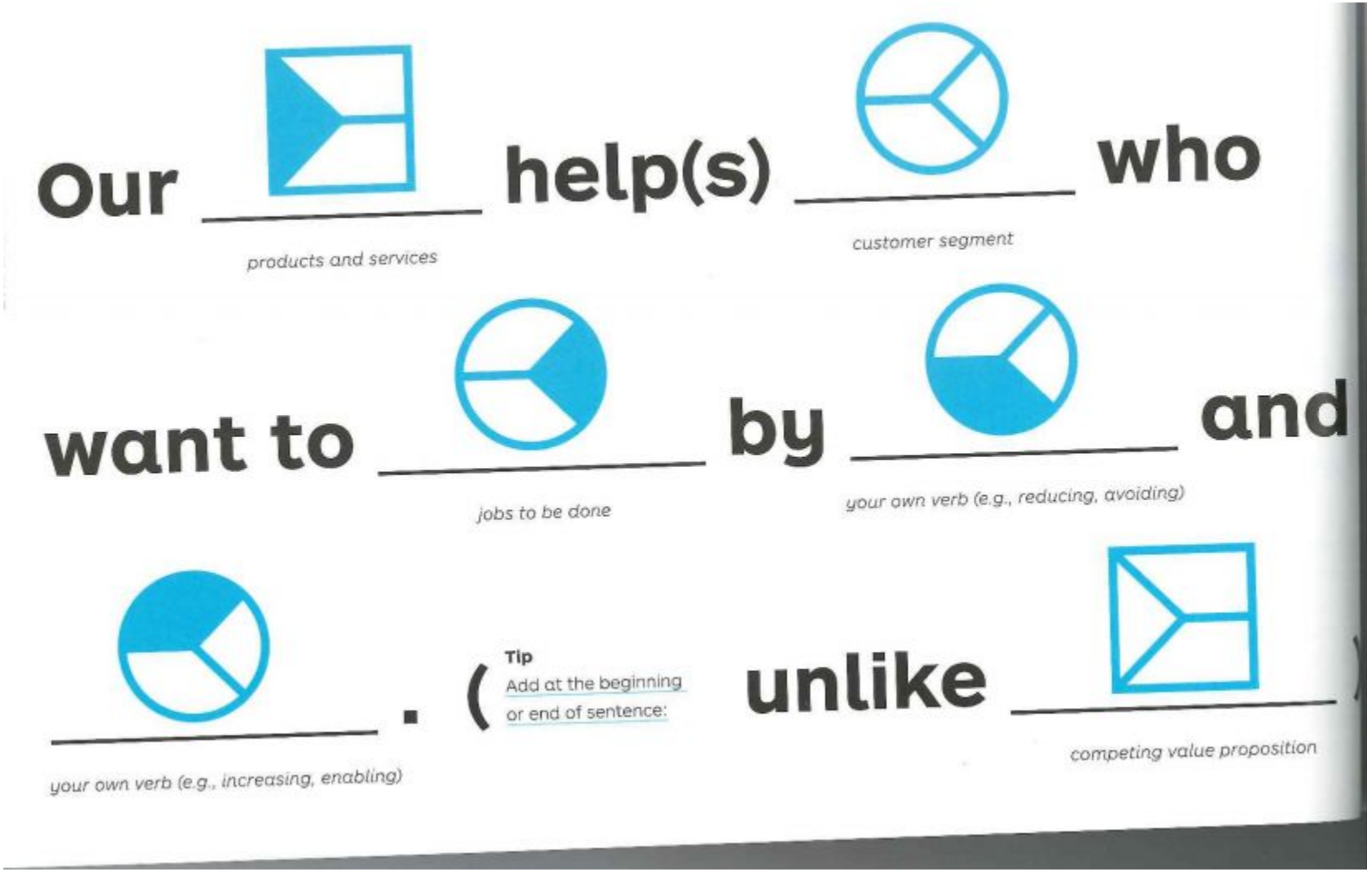
# E2E ML Project

## A valuable Data Scientist is a Business Developer with superpowers



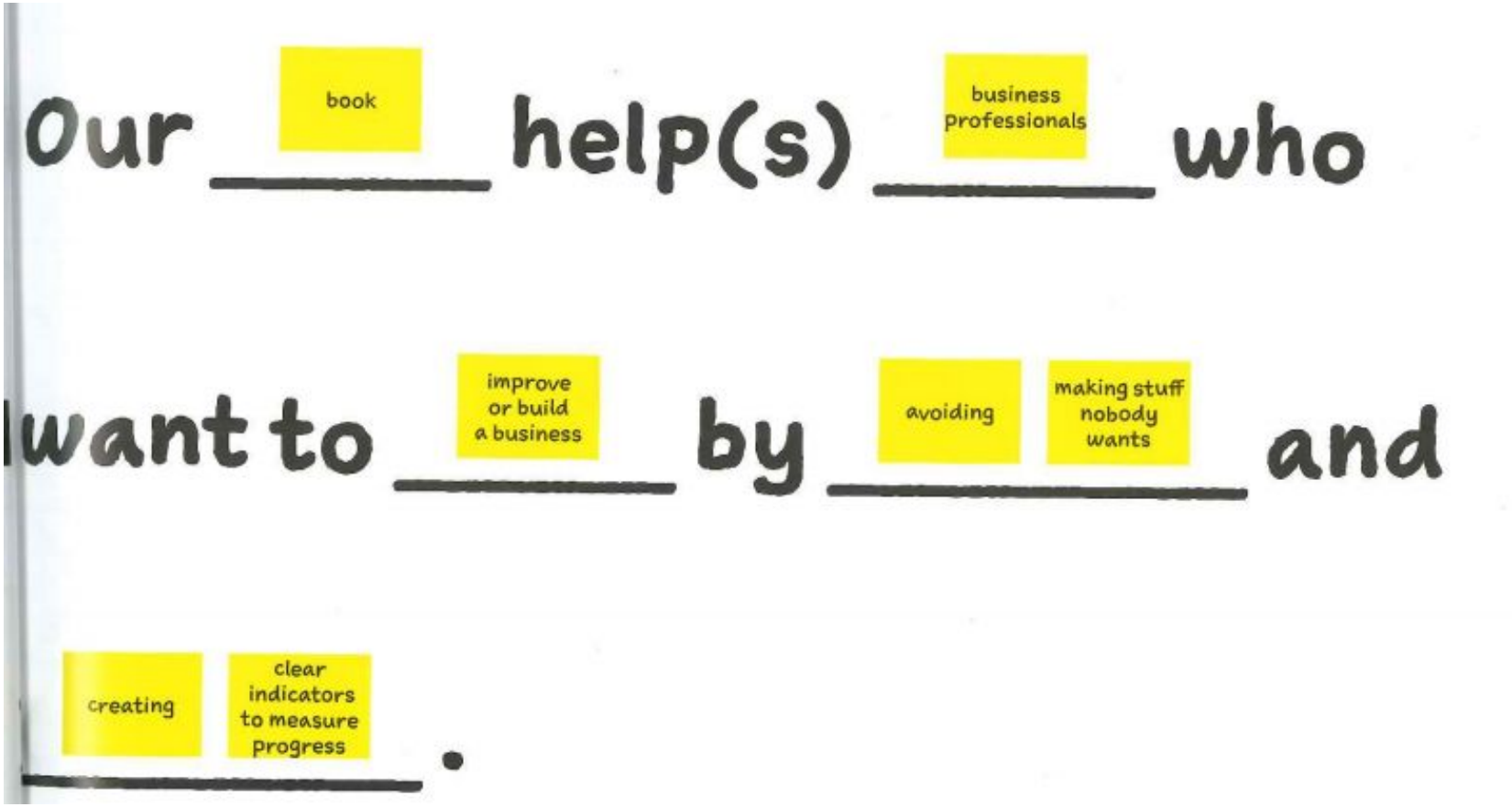
# E2E ML Project

## A valuable Data Scientist is a Business Developer with superpowers



# E2E ML Project

## A valuable Data Scientist is a Business Developer with superpowers



# E2E ML Project

**A valuable Data Scientist is a Business Developer with superpowers**

Your time now!

What is your business objective?

# End 2 “End” ML Project Checklist

# Get the Data

# E2E ML Project

## 2. Get the Data

Note: automate as much as possible so you can easily get fresh data.

1. List the data you need and how much you need.
2. Find and document where you can get that data.
3. Check how much space it will take.
4. Check legal obligations, and get authorization if necessary.
5. Get access authorizations.
6. Create a workspace (with enough storage space).
7. Get the data.
8. Convert the data to a format you can easily manipulate (without changing the data itself).
9. Ensure sensitive information is deleted or protected (e.g., anonymized).
10. Check the size and type of data (time series, sample, geographical, etc.).
11. Sample a test set, put it aside, and never look at it (no data snooping!).

# E2E ML Project

## Creating your workspace

- Depending on your working place, you will have already workspaces working for you
- Otherwise, I recommend to use Anaconda and follow the guidelines that we setup in this course
- Keep a requirements.txt file with the packages that your project needs
- Keep your environment clean!! Do not reuse environments and use one environment per project, thank me later.

# E2E ML Project

## Creating your workspace

Let’s go to the notebook

**UUID - #S12C1**

# End 2 “End” ML Project Checklist

# Explore the Data

# E2E ML Project

## 3. Explore the Data

Note: try to get insights from a field expert for these steps.

1. Create a copy of the data for exploration (sampling it down to a manageable size if necessary).
2. Create a Jupyter notebook to keep a record of your data exploration.
3. Study each attribute and its characteristics:
    1. Name
    2. Type (categorical, int/float, bounded/unbounded, text, structured, etc.)
    3. % of missing values
    4. Noisiness and type of noise (stochastic, outliers, rounding errors, etc.)
    5. Possibly useful for the task?
    6. Type of distribution (Gaussian, uniform, logarithmic, etc.)
4. For supervised learning tasks, identify the target attribute(s).Visualize the data.
5. Visualize the data.
6. Study the correlations between attributes.
7. Study how you would solve the problem manually.
8. Identify the promising transformations you may want to apply.
9. Identify extra data that would be useful
10. Document what you have learned.

# E2E ML Project

## Creating your workspace

Let’s go to the notebook

**UUID - #S12C2**

# End 2 “End” ML Project Checklist

# Prepare the Data

# E2E ML Project

## 4. Prepare the Data

- Work on copies of the data (keep the original dataset intact).
- Write functions for all data transformations you apply, for five reasons:
  - So you can easily prepare the data the next time you get a fresh dataset
  - So you can apply these transformations in future projects
  - To clean and prepare the test set
  - To clean and prepare new data instances once your solution is live
  - To make it easy to treat your preparation choices as hyperparameters

# E2E ML Project

## 4. Prepare the Data

1. Data cleaning:
    1. Fix or remove outliers (optional).
    2. Fill in missing values (e.g., with zero, mean, median…) or drop their rows (or columns).
2. Feature selection (optional):
    1. Drop the attributes that provide no useful information for the task.
3. Feature engineering, where appropriate:
    1. Discretize continuous features.
    2. Decompose features (e.g., categorical, date/time, etc.).
    3. Add promising transformations of features (e.g., log(x), sqrt(x), x^2, etc.).
    4. Aggregate features into promising new features.
4. Feature scaling: standardize or normalize features.

# E2E ML Project

## Creating your workspace

Let’s go to the notebook

**UUID - #S12C3**

# End 2 “End” ML Project Checklist

# Short-List Promising Models

# E2E ML Project

## 5. Short-List Promising Models

- If the data is huge, you may want to sample smaller training sets so you can train many different models in a reasonable time (be aware that this penalizes complex models such as large neural nets or Random Forests).

- Once again, try to automate these steps as much as possible.

# E2E ML Project

## 5. Short-List Promising Models

1. Train many quick and dirty models from different categories (e.g., linear, naïve Bayes, SVM, Random Forests, neural net, etc.) using standard parameters.
2. Measure and compare their performance.
    1. For each model, use *N*-fold cross-validation and compute the mean and standard deviation of the performance measure on the *N* folds.
3. Analyze the most significant variables for each algorithm.
4. Analyze the types of errors the models make.
    1. What data would a human have used to avoid these errors?
5. Have a quick round of feature selection and engineering.
6. Have one or two more quick iterations of the five previous steps.
7. Short-list the top three to five most promising models, preferring models that make different types of errors.

# E2E ML Project

## Creating your workspace

Let’s go to the notebook

**UUID - #S12C4**

# End 2 “End” ML Project Checklist

# Fine-Tune the System

# E2E ML Project

## 6. Fine-Tune the system

- You will want to use as much data as possible for this step, especially as you move toward the end of fine-tuning.

- As always automate what you can.

# E2E ML Project

## 6. Fine-Tune the system

1. Fine-tune the hyperparameters using cross-validation.
    1. Treat your data transformation choices as hyperparameters, especially when you are not sure about them (e.g., should I replace missing values with zero or with the median value? Or just drop the rows?).
    2. Unless there are very few hyperparameter values to explore, prefer random search over grid search. If training is very long, you may prefer a Bayesian optimization approach
2. Try Ensemble methods. Combining your best models will often perform better than running them individually.
3. Once you are confident about your final model, measure its performance on the test set to estimate the generalization error.
4. Don’t tweak your model after measuring the generalization error: you would just start overfitting the test set.

# E2E ML Project

## Creating your workspace

Let’s go to the notebook

**UUID - #S12C5**

# End 2 “End” ML Project Checklist

# Present your solution

# E2E ML Project

## 6. Present your Solution

1. Document what you have done.
2. Create a nice presentation.
    1. Make sure you highlight the big picture first.
3. Explain why your solution achieves the business objective.
4. Don’t forget to present interesting points you noticed along the way.
    1. Describe what worked and what did not.
    2. List your assumptions and your system’s limitations.
5. Ensure your key findings are communicated through beautiful visualizations or easy-to-remember statements (e.g., “the median income is the number-one predictor of housing prices”).

# End 2 “End” ML Project Checklist

# Launch your solution

# E2E ML Project

## 7. Launch your solution

1. Get your solution ready for production (plug into production data inputs, write unit tests, etc.).
2. Write monitoring code to check your system's live performance at regular intervals and trigger alerts when it drops.
    1. Beware of slow degradation too: models tend to "rot" as data evolves.
    2. Measuring performance may require a human pipeline (e.g., via a crowdsourcing service).
3. Also monitor your inputs' quality (e.g., a malfunctioning sensor sending random values, or another team's output becoming stale). This is particularly important for online learning systems.
4. Retrain your models on a regular basis on fresh data (automate as much as possible).

# E2E ML Project

## 7. Launch your solution

In ADA we will have a fully fledged launched solution using Sklearn pipelines for data preparation, joblib for model persistence, DVC for versioning, MLFlow for experimentation and AWS for launch!

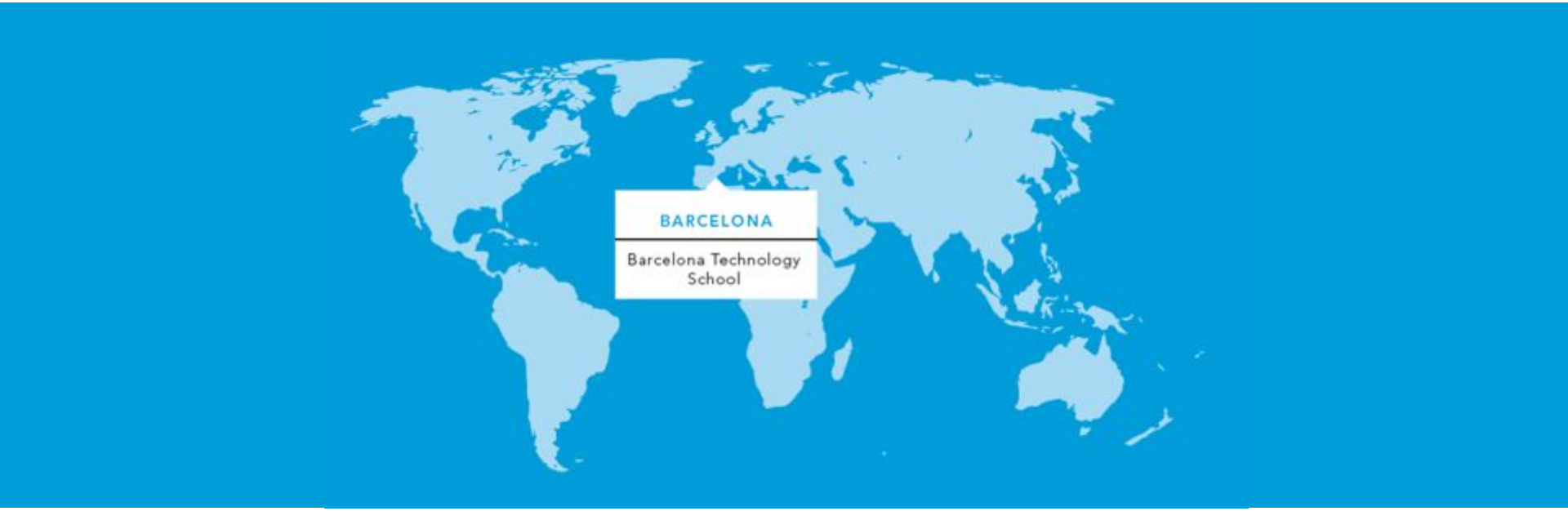
Also in the following days we will see how to publish a model using heroku and streamlit

# RECAP

# Resources

## Important resources

- Lex Friedman Series (MIT)
- Sklearn docs
- Jake VanderPlas, “Python Data Science Handbook”
- Aurelien Geron, “Hands-on machine learning with scikit-learn, Keras & Tensorflow”


BARCELONA
Barcelona Technology School