UNIVERSITY OF TORONTO
SCHOOL OF CONTINUING STUDIES

3253 Analytic Techniques and Machine Learning

Module 2: End to End Machine Learning Project

Course Plan

Module Titles
Module 1 – Introduction to Machine Learning
Current Focus: Module 2 – End to End Machine Learning Project
Module 3 – Classification
Module 4 – Clustering and Unsupervised Learning
Module 5 – Training Models and Feature Selection
Module 6 – Support Vector Machines
Module 7 – Decision Trees and Ensemble Learning
Module 8 – Dimensionality Reduction
Module 9 – Introduction to TensorFlow
Module 10 – Introduction to Deep Learning and Deep Neural Networks
Module 11 – Distributing TensorFlow, CNNs and RNNs
Module 12 – Final Assignment and Presentations (no content)

Learning Outcomes for this Module

- Perform all of the steps of building a simple machine learning project:
  - Conduct exploratory analysis and visualization
  - Prepare data
  - Set aside a testing set
  - Select and train a model
  - Deploy, monitor and maintain it

Topics for this Module

- **2.1** Loss functions
- **2.2** Feature scaling
- **2.3** Test sets
- **2.4** Geographical data
- **2.5** Correlation
- **2.6** Encoding features
- **2.7** Cross-validation
- **2.8** Searching for hyperparameters
- **2.9** Running the application
- **2.10** Resources and Wrap-up

Module 2 – Section 1

Loss Functions

Notations

- m is the number of instances in dataset
- $\boldsymbol{x}^{(1)} = \begin{pmatrix} -118.29 \\ 33.91 \\ 1416 \\ 38{,}372 \end{pmatrix}$
- $\mathbf{X}$ is a matrix containing all feature values
- $\mathbf{X} = \begin{pmatrix} \left(x^{1}\right)^{T} \\ \left(x^{(2)}\right)^{T} \\ \cdots \\ \left(x^{100}\right)^{T} \end{pmatrix} = \begin{pmatrix} -118.29 & 33.91 & 1416 & 38,372 \\ \vdots & \vdots & \vdots & \vdots \end{pmatrix}$

Predictions

- *h* is system's prediction function, aka a *hypothesis*
- When system is given an input $\mathbf{x}^{(i)}$, it outputs a predicted value y_hat(i) = h($x^{(i)}$)
- For a regression task, will often use Root Mean Squared Error (RMS-E) or Mean Absolute Error (MAE) as the cost function

Performance Measures

- Root Mean Square Error (RMS-E) measures standard deviation c of errors in prediction
- $RMSE(\boldsymbol{X}, h) = \sqrt{\frac{1}{m}\sum_{i=1}^{m}\left(h\left(x^{(i)}\right) - y^{(i)}\right)^2}$
- In housing example, assuming a normal distribution, σ=50,000 implies that
  – ~68% of system's predictions fall within $50,000 of actual value
  – ~95% of predictions fall within $100,000 of actual value

Performance Measures (cont'd)

Mean Absolute Error (MAE) measures standard deviation c of errors in prediction

$$MAE(X, h) = \frac{1}{m} \mid \mathrm{h}\left(x^{i}\right) - \ y^{i}) \mid$$

UNIVERSITY OF TORONTO
SCHOOL OF CONTINUING STUDIES

Module 2 – Section 2

Feature Scaling

Feature Scaling

- ML algorithms usually perform poorly when input features have different scales
- Normalization or min-max scaling:
  - values are shifted and rescaled so result is in range [0,1]

$$x_{scaled} = \frac{x - x_{min}}{x_{max} - x_{min}}$$

Example: If X = [0, 50, 100], what is X_{scaled}?

Feature Scaling

- ML algorithms usually perform poorly when input features have different scales
- Normalization or min-max scaling:
  - values are shifted and rescaled so result is in range [0,1]

$$x_{scaled} = \frac{x - x_{min}}{x_{max} - x_{min}}$$

Example: If X = [0, 50, 100], what is X_{scaled}?

$0_{scaled} = \frac{0-0}{100-0}$ = 0, $50_{scaled} = \frac{50-0}{100-0}$ = 0.5, $100_{scaled} = \frac{100-0}{100-0}$ = 1

Normalization in Python

```
x = [[0],[50],[100]]
x
```

```
[[0], [50], [100]]
```

```
#import MinMaxscaler
from sklearn.preprocessing import MinMaxScaler
#create scaler object
scaler = MinMaxScaler()
# fit scaler object
scaler.fit(x)

#now, we can transform x using the trained/fitted scaler
x_scaled = scaler.transform(x)
print(x_scaled)
```

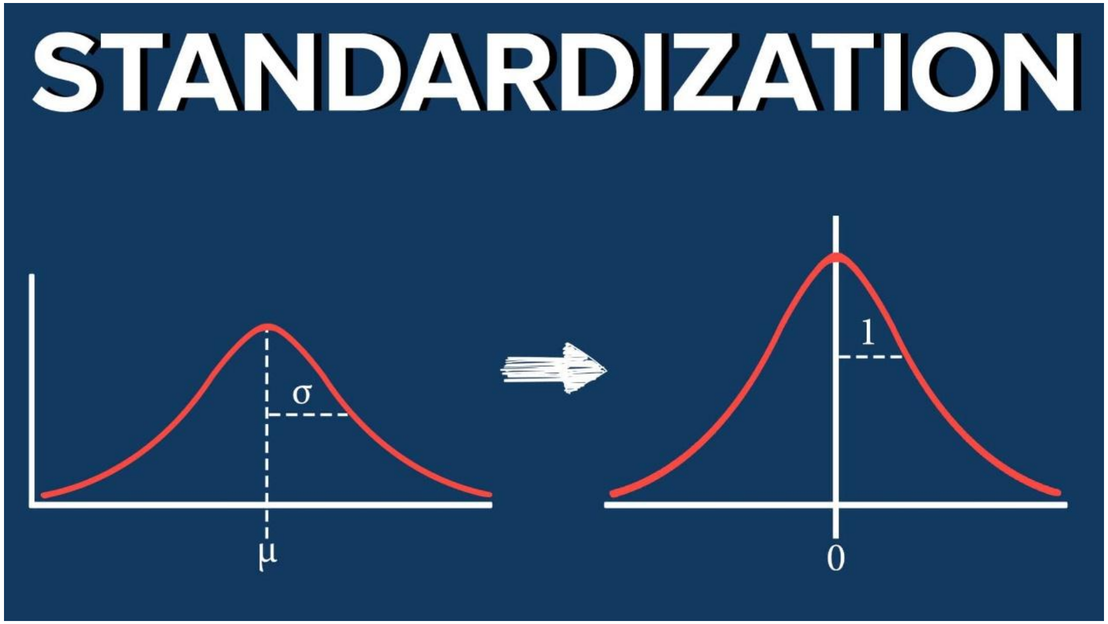
```
[[0. ]
 [0.5]
 [1. ]]
```

Review code in “Module 2 - Standardization code.ipynb”

Standardization

- Standardization
  - Two operations:
    - Subtract mean (giving zero mean)
    - Divide by variance (unit mean)

$$Z = \frac{x_i - \mu}{\sigma}, \quad \boldsymbol{\mu} \text{ is mean } \& \ \boldsymbol{\sigma} \text{ is variance}$$



Standardization

Pay attention to *fit_transform()* function. Is it combination of fit and transform?

```
x = [[0],[50],[100]]
x
```

```
[[0], [50], [100]]
```

```
from sklearn.preprocessing import StandardScaler

StandardScaler = StandardScaler()
x_ss = StandardScaler.fit_transform(x)
x_ss
```

```
array([[-1.22474487],
       [ 0.        ],
       [ 1.22474487]])
```

What is mean and sd of x_ss?

```
import numpy
numpy.std(x_ss)
```

```
0.9999999999999999
```

Normalization vs Standard Scaler

- Standardization does not bound values to a specific range
- Can be problematic for certain ML methods — neural nets work well with input range [0,1]
- Much less affected by outliers than min-max
  - one large value will “crush” all other values into a very limited range

UNIVERSITY OF TORONTO
SCHOOL OF CONTINUING STUDIES

Module 2 – Section 3

Test Sets

Create a Test Set

- ***Data snooping bias*** -- studying the test set might result in observing some interesting pattern, which when accounted for in model means that model will not generalize as well
- Pick some instances randomly from dataset randomly and set them aside
- Typically 20% of the dataset
- Important to set aside subset before training starts, don't randomly generate each run
  - over time, your algorithm will see whole dataset, which will invalidate the whole validation process

Create a Test Set (cont'd)

- Pick some instances randomly from dataset randomly and set them aside
- Typically 20% of the dataset
- Important to set aside subset before training starts, don't randomly generate each run
  - over time, your algorithm will see whole dataset, which will invalidate the whole validation process

Create a Test Set (cont'd)

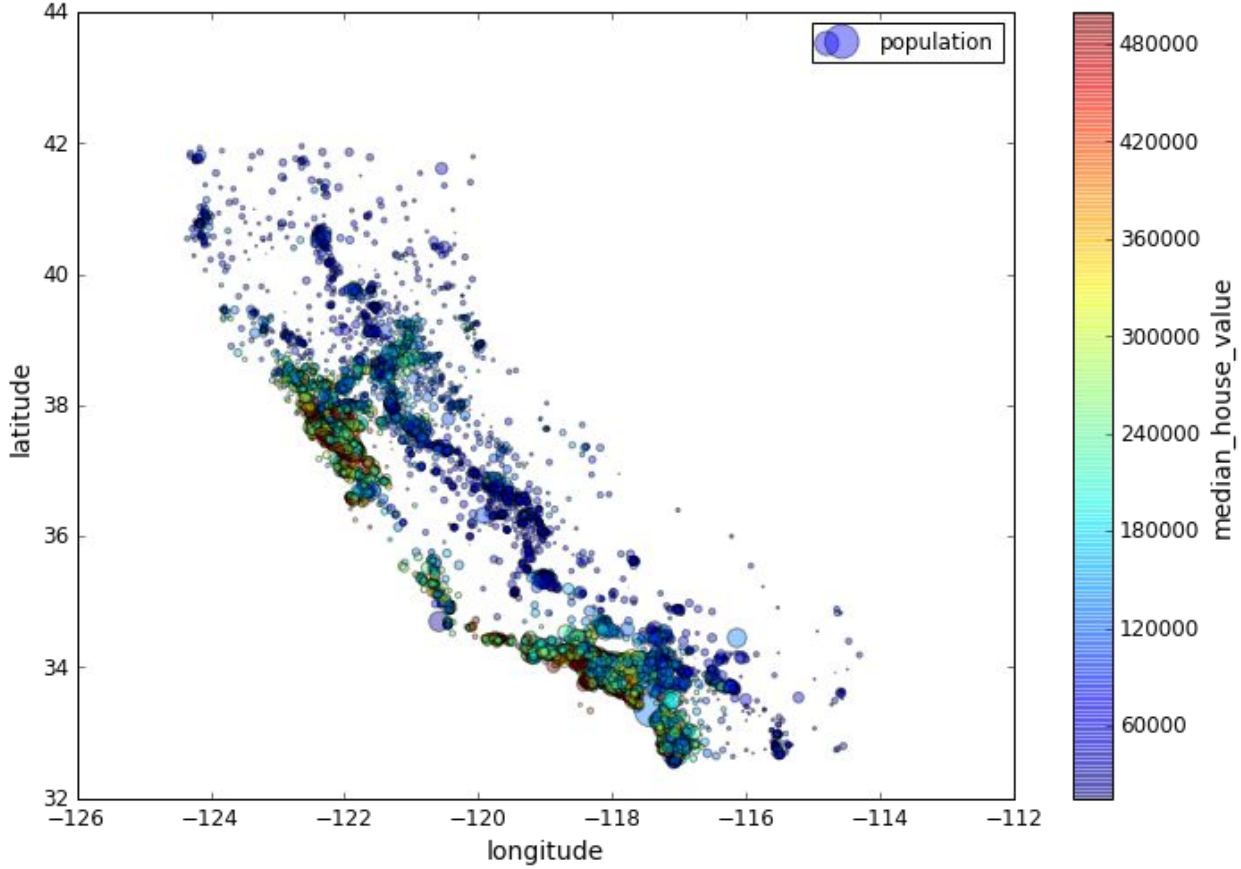
- Using random sampling is usually acceptable if dataset is sufficiently large with respect to number of attributes
- For small datasets, purely random sampling may not work
- Example:
  - Population of USA composed of 51.3% females, 48.7% males
  - ensuring that a survey of the population maintains this ratio is referred to as *stratified sampling*

UNIVERSITY OF TORONTO
SCHOOL OF CONTINUING STUDIES

Module 2 – Section 4

Geographical Data

Visualizing Geographical Data



UNIVERSITY OF TORONTO
SCHOOL OF CONTINUING STUDIES

Visualizing Geographical Data (cont'd)

- As seen in previous image, strong correlation between population size and income
- Also related to distance to the coast
- Plotting data can be extremely useful to gain insights to test
- Scatterplots (as seen previously) can be helpful, histograms too
- Simple plotting may expose a relationship between variables

UNIVERSITY OF TORONTO
SCHOOL OF CONTINUING STUDIES

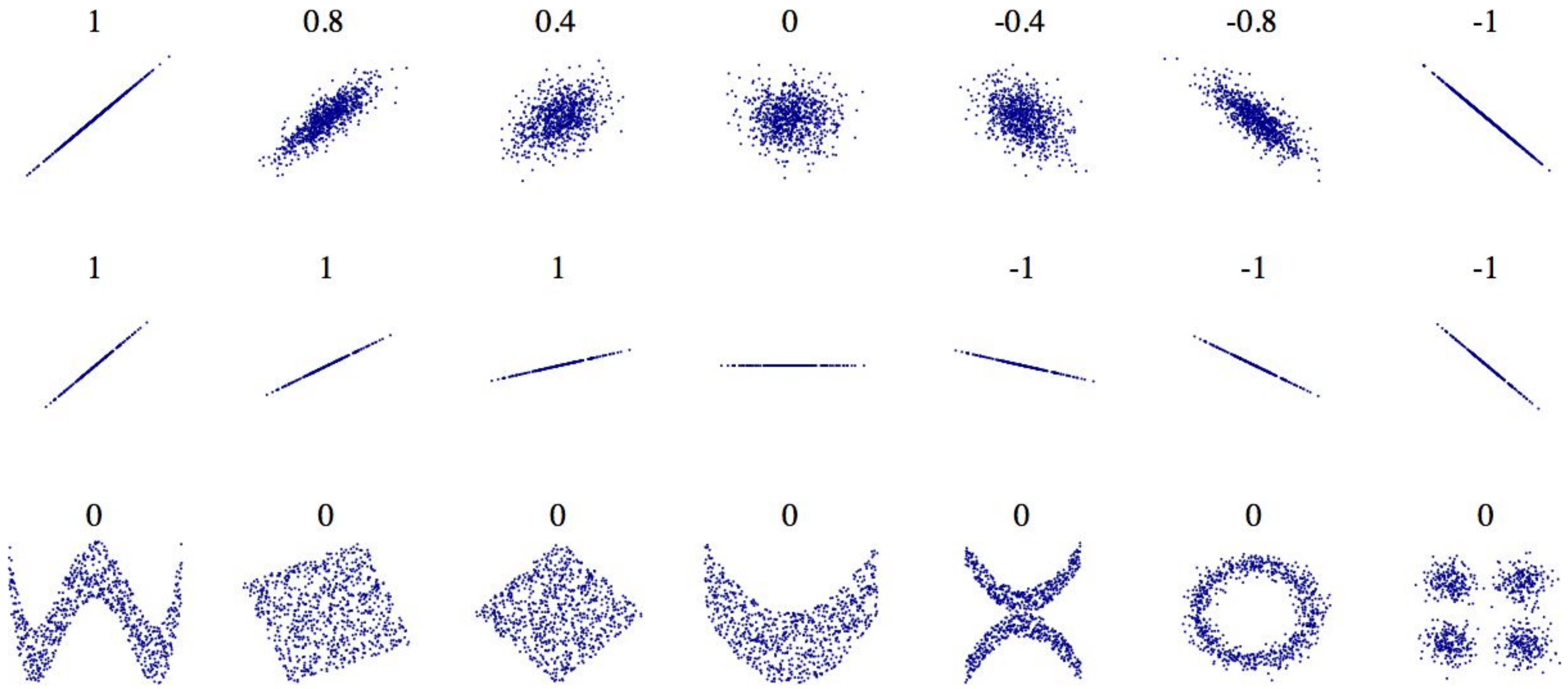
Module 2 – Section 5

Correlation

Correlation Coefficient

- Correlation is measure of the degree to which two variables have a linear relationship between them
- Correlation coefficient varies between [-1.0,1.0]
- Note that correlation coefficient only measures *linear* relationship between variables
- Pearson's correlation coefficient $\rho_{x,y} = \frac{cov(x,y)}{\sigma_x \sigma_y}$

Correlation Coefficient (cont'd)



Module 2 – Section 6

Encoding Features

One-Hot-Encoding

How to deal with categorical features? Can I multiply 'Human' by a coefficient?

Sample	Category
1	Human
2	Human
3	Penguin
4	Octopus
5	Alien
6	Octopus
7	Alien

https://www.quora.com/What-is-one-hot-encoding-and-when-is-it-used-in-data-science

One-Hot-Encoding

How to deal with categorical features? Can I multiply 'Human' by a coefficient?

Sample	Category	Numerical
1	Human	1
2	Human	1
3	Penguin	2
4	Octopus	3
5	Alien	4
6	Octopus	3
7	Alien	4

It looks working, right?

Do you have concern?

https://www.quora.com/What-is-one-hot-encoding-and-when-is-it-used-in-data-science

One-Hot-Encoding

How to deal with categorical features? Can I multiply 'Human' by a coefficient?

Sample	Category	Numerical
1	Human	1
2	Human	1
3	Penguin	2
4	Octopus	3
5	Alien	4
6	Octopus	3
7	Alien	4

It looks working, right?

Do you have concern?

2 - 1 = 3 - 2
"Penguin" vs "Human" = "Octopus" vs "Penguin"

4 - 1 = 3 * (4 - 3)
"Alien" vs "Human". = 3 * "Alien" vs "Octopus"

https://www.quora.com/What-is-one-hot-encoding-and-when-is-it-used-in-data-science

One-Hot-Encoding

Let's create dummy variables then!

Sample	Category	Numerical
1	Human	1
2	Human	1
3	Penguin	2
4	Octopus	3
5	Alien	4
6	Octopus	3
7	Alien	4

Sample	Human	Penguin	Octopus	Alien
1	1	0	0	0
2	1	0	0	0
3	0	1	0	0
4	0	0	1	0
5	0	0	0	1
6	0	0	1	0
7	0	0	0	1

https://www.quora.com/What-is-one-hot-encoding-and-when-is-it-used-in-data-science

Curse of Dummy Variable

Let’s create dummy variables then!

Sample	Category	Numeric al	Human	Penguin	Octopus	Alien
1	Human	1	1	0	0	0
3	Penguin	2	0	1	0	0
4	Octopus	3	0	0	1	0
5	Alien	4	0	0	0	1

Curse of Dummy Variable

Let's create dummy variables then!

Sample	Category	Numeric al	Human	Penguin	Octopus	Alien
1	Human	1	1	0	0	0
3	Penguin	2	0	1	0	0
4	Octopus	3	0	0	1	0
5	Alien	4	0	0	0	1

You should always drop one of the dummy variables, otherwise your dataset is cursed with excessive feature that has no value!!!

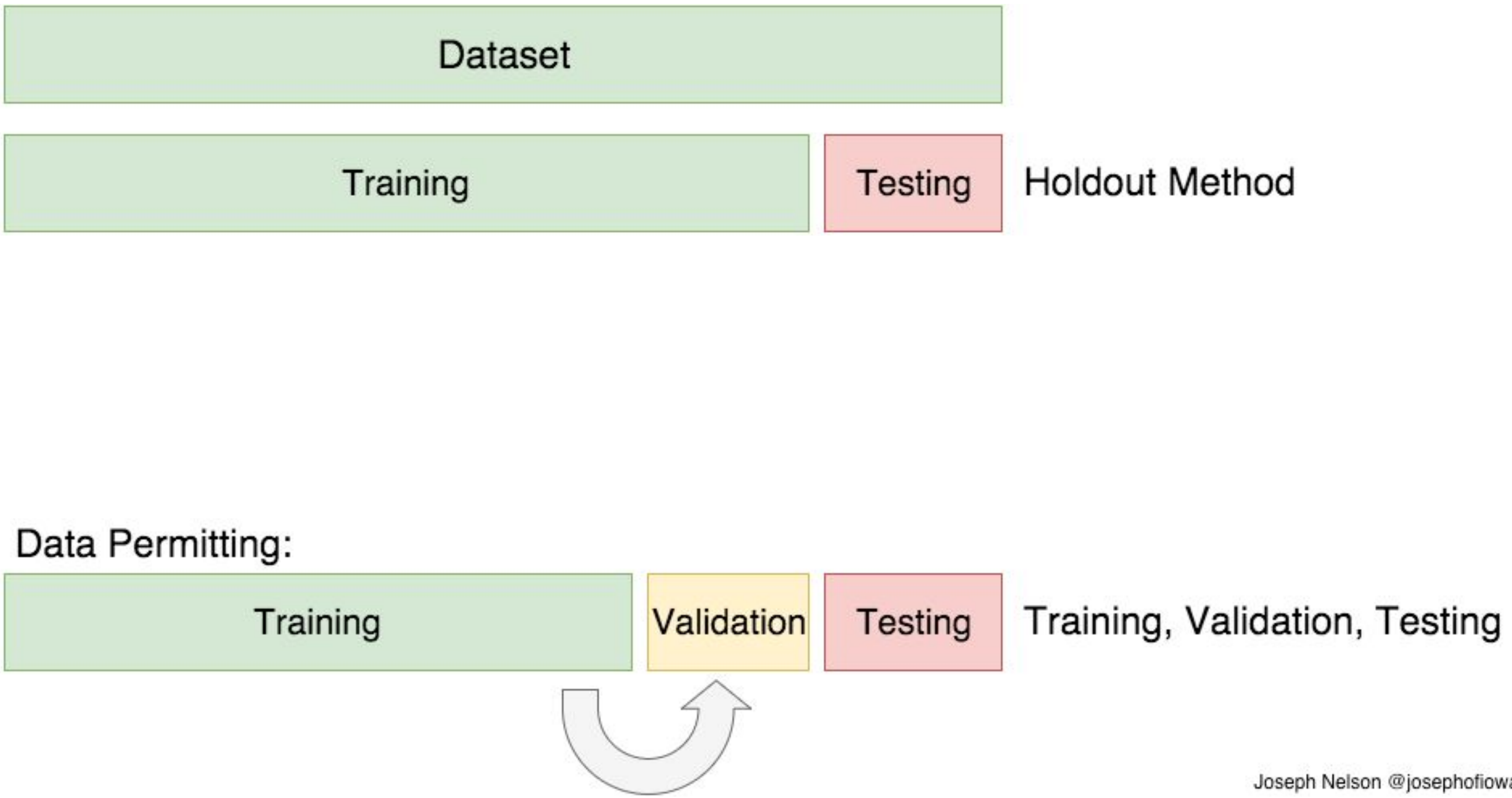
UNIVERSITY OF TORONTO
SCHOOL OF CONTINUING STUDIES

Module 2 – Section 7

Cross-Validation

Train, Test, Validation: Recap

Train, Test, Validation: Recap



Cross Validation

- As mentioned, splitting in train/test is insufficient and we are tempted to break it down into train/valid/test
- Splitting to train/valid/test crunches training dataset size if dataset is small
- Also, how can I prove that my model is not great on a cherry picked train/test dataset?

Solution ?

- Use cross-validation

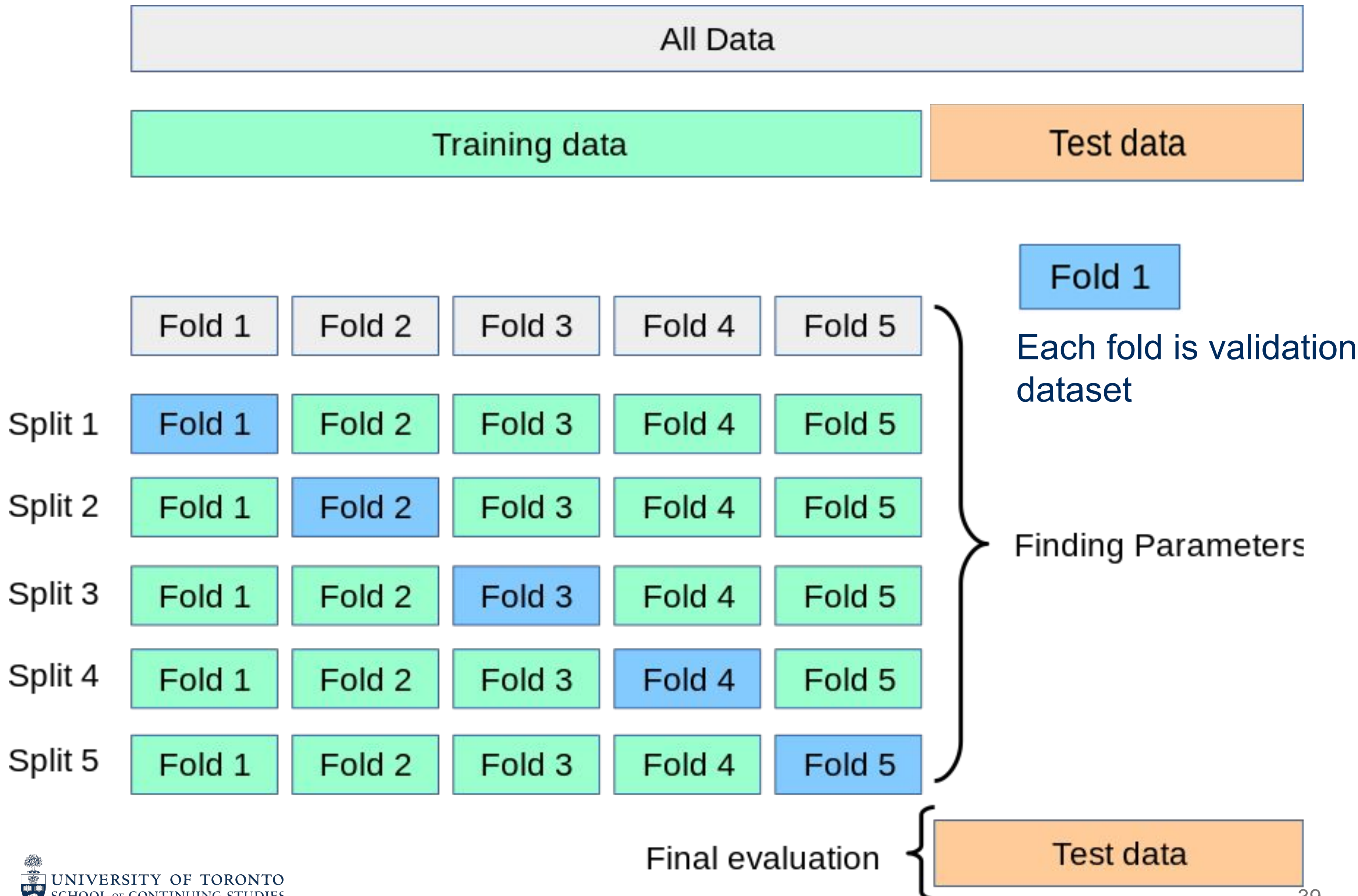
Cross Validation

K-fold cross validation — randomly split **training** data into k distinct subsets called *folds*

Trains and evaluates model k times, picking different fold each time for validation and training on the k-1 other folds

Allows for estimate of model performance, but also estimate of how precise this estimate is (standard deviation)

Example with K = 5 (usually K can be 3, 5, 10)



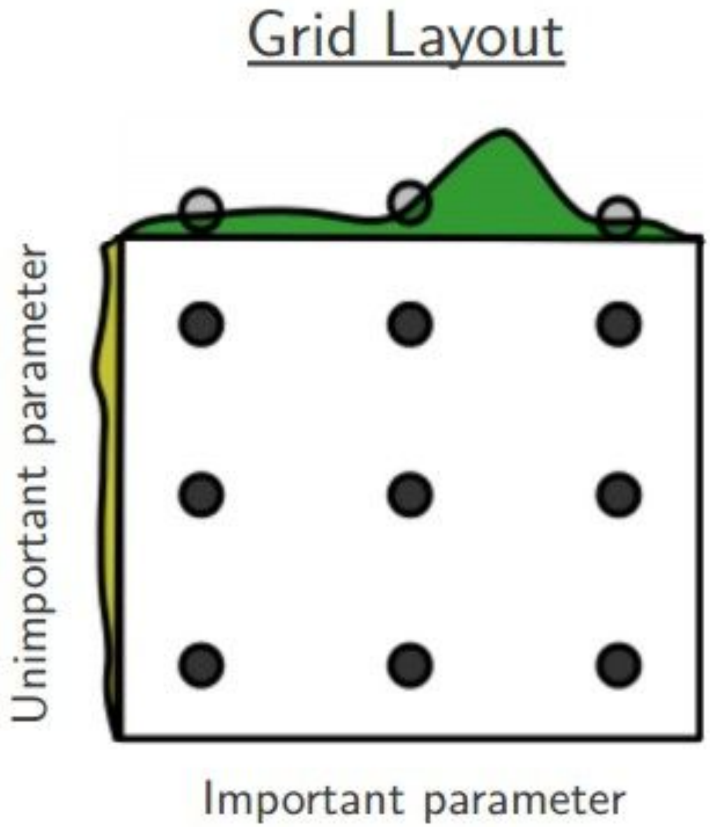
Module 2 – Section 8

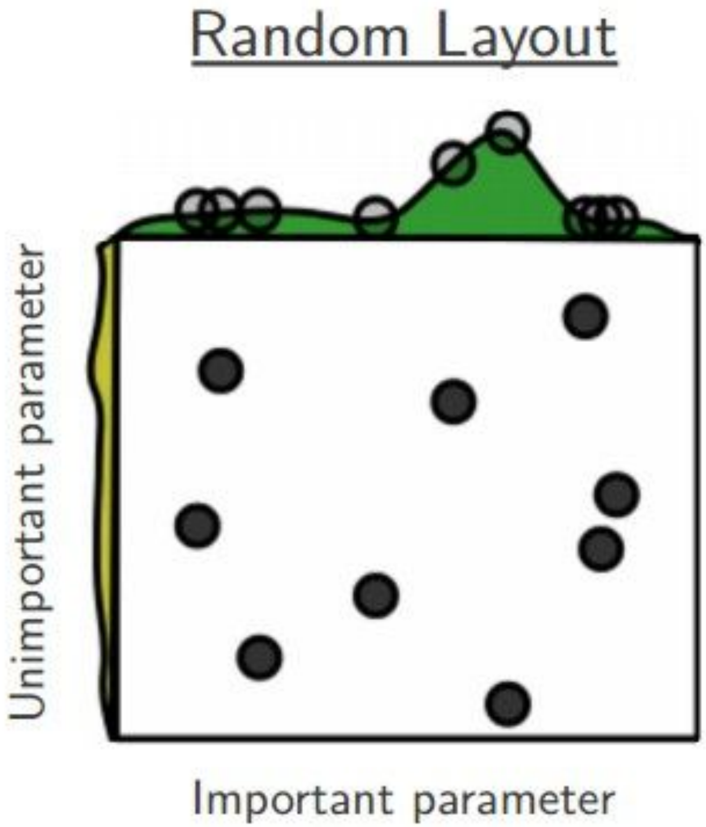
Searching for Hyperparameters

Searching for Hyperparameters

- Once a basic model has been trained, need to search for **optimal hyperparameters** (learning rate, weights to balance loss terms, etc)
- ***Randomized search***: useful for large search spaces
- Evaluates a given number of random combinations by selecting a random value for each hyperparameter at each iteration
- **Grid search:** try every combination of the sets of hyperparameters provided

Grid Search


Grid Layout
Unimportant parameter
Important parameter


Random Layout
Unimportant parameter
Important parameter

Grid Search (cont'd)

- Each dict has keys that correspond to args for the model
- Associated dict is an exhaustive list of params to be tested
- Grid search finds best set of params

```
param_grid = [
    {'n_estimators':[3, 10, 30],
     'max_features':[2, 4, 6, 8]},
    {'bootstrap':[False],
     'n_estimators': [3, 10],
     'max_features':[2, 3, 4]}]
forest_reg =
RandomForestRegressor()
grid_search =
GridSearchCV(forest_reg,
    param_grid,
     cv=5,
   scoring='neg_mean_squared_error')
```

Module 2 – Section 9

Running the Application

Launch, Monitor and Maintain System

- Need to write code to **monitor performance**
- Not just monitoring for breakage, but ability of model to make accurate predictions as data evolves (“data rot”)
- **Make sure to monitor quality of inputted data:** performance of model may degrade slowly
- Catching issues like **malfunctioning sensor input** or stale data from another team will allow for faster resolution of issues

Module 2 – Section 10

Resources and Wrap-up

Homework

- Complete the notebook in the assignments section for this week

Next Class

- Classification
- Reading: Chapter 3

Follow us on social

Join the conversation with us online:

facebook.com/uoftscs

@uoftscs

linkedin.com/company/university-of-toronto-school-of-continuing-studies

@uoftscs

Any questions?

Thank You

Thank you for choosing the University of Toronto School of Continuing Studies