LECTURE 3 TERM 2:

MSIN0097

UCL SCHOOL OF MANAGEMENT

PREDICTIVE ANALYTICS

A P MOORE

INDIVIDUAL COURSEWORK

UCL SCHOOL OF MANAGEMENT

MSIN0097

PREDICTIVE ANALYTICS
INDIVIDUAL COURSEWORK

A P MOORE

1

INDIVIDUAL COURSEWORK

— Friday 26th February 2021
— 60% of module mark
— 2000 words

2

BRIEF

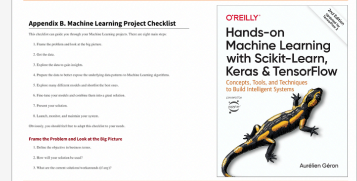
The individual coursework task is to identify a dataset and explore building a predictive model using the methods and techniques presented in the first 5 weeks of the course.

There are six main steps:

1. Obtain a dataset and explain the problem you are trying to solve.
   This will characterise the type of predictive model you can build
2. Explore the data to gain insights.
   Visualize and explain the main trends in the data
3. Prepare the data to better expose the underlying data patterns to Machine Learning algorithms.
4. Explore different models and shortlist the best ones.
5. Fine-tune your models and combine them into a great solution.
6. Present your final solution with any summary conclusions.

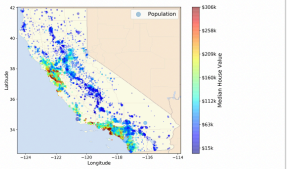
3

GUIDANCE



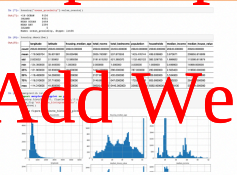
4

END-TO-END



5

NOTEBOOK



6

DATASETS

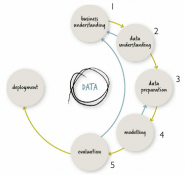
Useful places for ML datasets:

— Tabular & cleaned: https://github.com/EpistasisLab/pmlb/tree/master/datasets
— By domain: https://datasetlist.com
— By application: https://github.com/awesomedata/awesome-public-datasets
— Search engine: https://datasetsearch.research.google.com

7

CRISP CYCLE

DATA DEVELOPMENT LIFECYCLES



8

GUIDANCE

~~Get~~ First Pass

Make a first-pass through the project steps as fast as possible.This will give you confidence that you have all the parts that you need and a baseline from which to improve.

Cycles - The process in not linear but cyclic. You will loop between steps, and probably spend most of your time in tight loops between steps 3-4 or 3-4-5 until you achieve a level of accuracy that is sufficient or you run out of time.

The write up in the final submitted Notebook can be more linear - you do not need to include all of your work, ie. including all dead-ends, and it should be concise and consistent.

9

GUIDANCE

Attempt Every Step

It is easy to skip steps, especially if you are not confident or familiar with the tasks of that step.Try and do something at each step in the process, even if it does not improve accuracy.You can always build upon it later.Don't skip steps, just reduce their contribution to your final submission as necessary.

Ratchet Accuracy

The goal of the project is to achieve good model performance (which ever metric you use to measure this). Every step contributes towards this goal.

Set some simple benchmarks early on.Treat changes that you make as experiments that potentially increase accuracy.

Performance is a ratchet that can only move in one direction (better, not worse).

10

GUIDANCE

Adapt As Needed

Modify the steps as you need on a project, especially as you become more experienced with using the Notebook.

The final submitted Notebook does not need to preserve the suggested structure if you think something else is more appropriate.

11

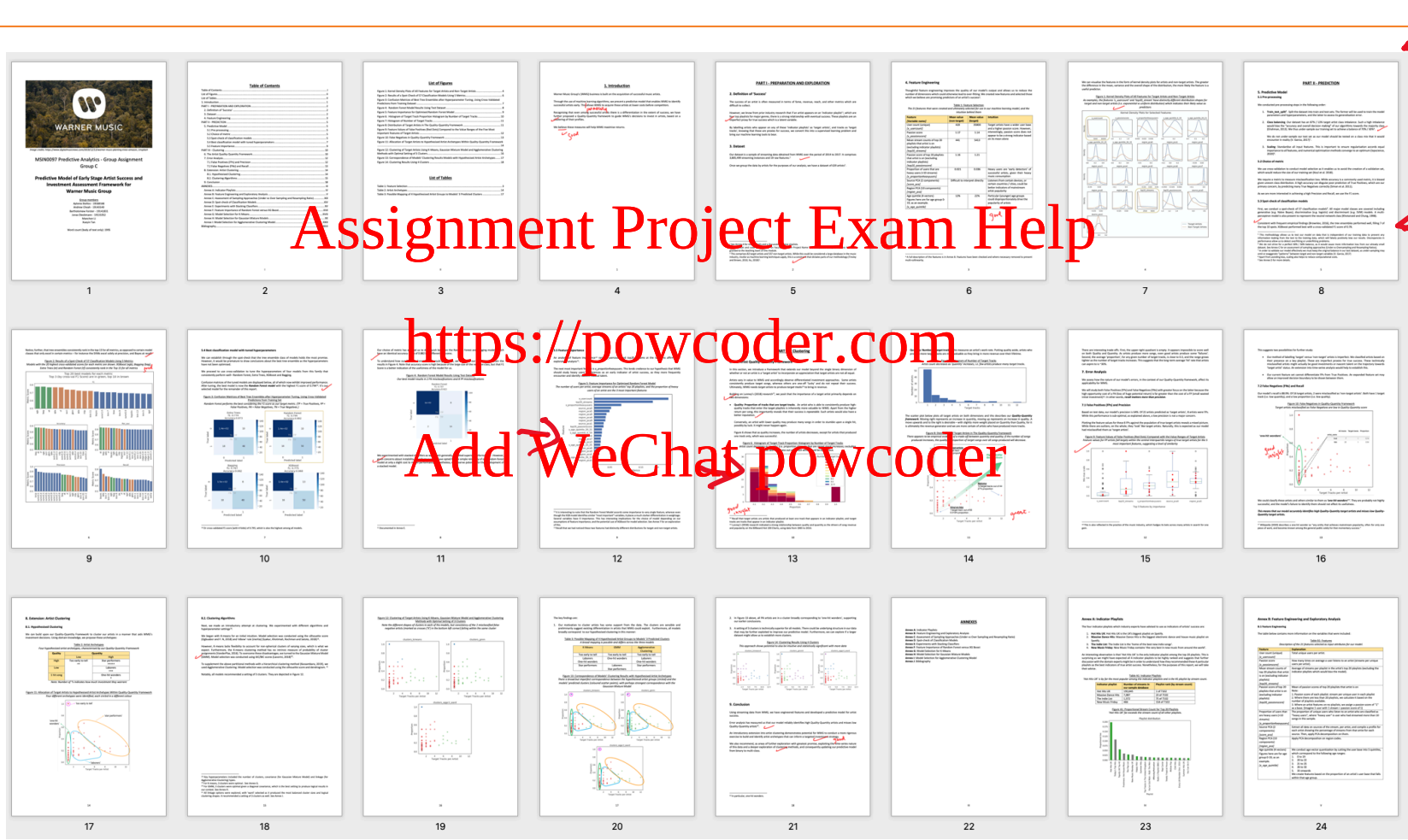
KEY DATES

— Submission Friday 26th February 2021, 10 am

12

COURSEWORK / INDUSTRY REPORT

UCL SCHOOL OF MANAGEMENT



MACHINE LEARNING JARGON

UCL SCHOOL OF MANAGEMENT

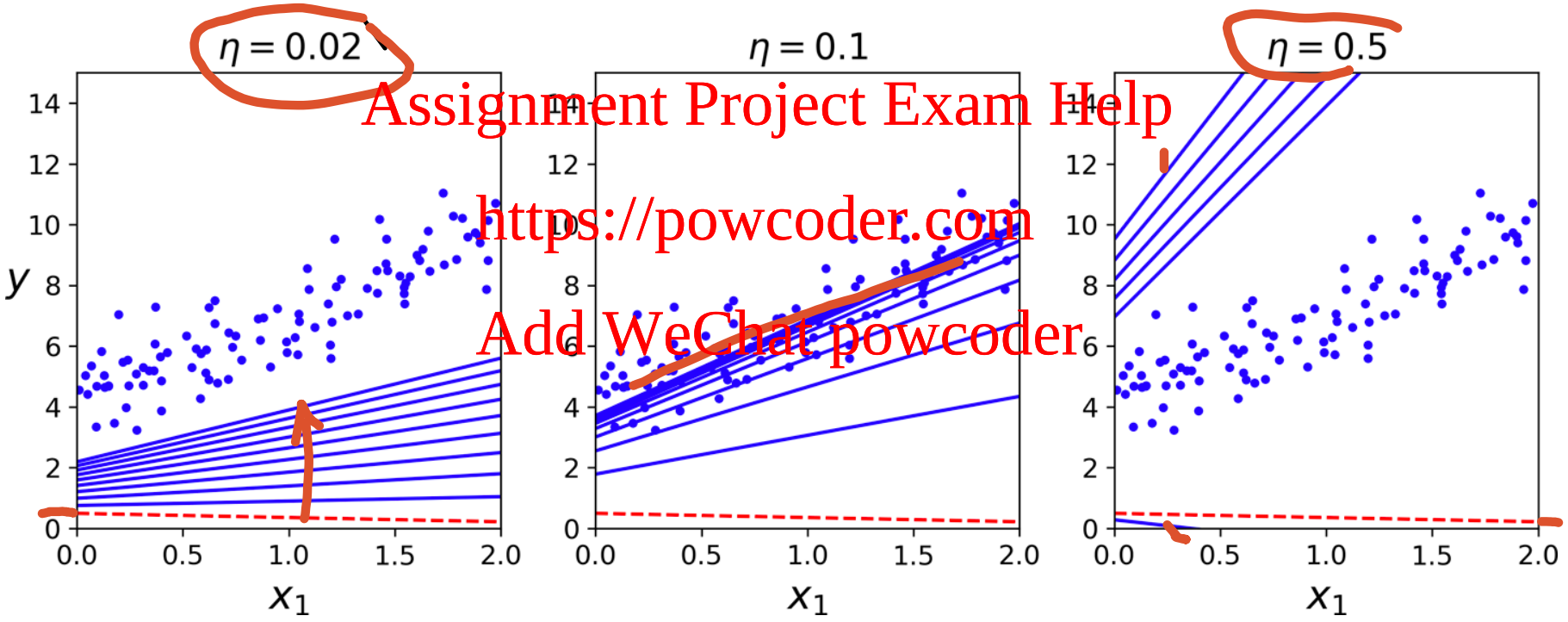
- Model
- Interpolating / Extrapolating
- Data Bias
- Noise / Outliers
- Learning algorithm
- Inference algorithm
- **Supervised learning**
- Unsupervised learning
- **Classification**
- **Regression**
- Clustering
- Decomposition

- Parameters
- Optimisation
- Training data
- Testing data
- Error metric
- Linear model
- Parametric model
- Model variance
- Model bias
- Model generalization
- Overfitting
- Goodness-of-fit
- Hyper-parameters

- Failure modes
- Confusion matrix
- True Positive
- False Negative
- Partition
- Data density
- Hidden parameter
- High dimensional space
- Low dimensional space
- Separable data
- Manifold / Decision surface
- Hyper cube / volume / plane

LEARNING RATES

UCL
SCHOOL OF
MANAGEMENT



Hands of ML: Chapt 4, Figure 4.8, page 116 (v1)

CODE

UCL SCHOOL OF MANAGEMENT

```
theta_path_bgd = []

def plot_gradient_descent(theta, eta, theta_path=None):
    m = len(X_b)
    plt.plot(X, y, "b.")
    n_iterations = 1000
    for iteration in range(n_iterations):
        if iteration < 10:
            y_predict = X_new_b.dot(theta)
            style = "b-" if iteration > 0 else "r--"
            plt.plot(X_new, y_predict, style)
        gradients = 2/m * X_b.T.dot(X_b.dot(theta) - y)
        theta = theta - eta * gradients
        if theta_path is not None:
            theta_path.append(theta)
    plt.xlabel("$x_1$", fontsize=18)
    plt.axis([0, 2, 0, 15])
    plt.title(r"$\eta = {}$".format(eta), fontsize=16)
```

ALTERING CODE

UCL SCHOOL OF MANAGEMENT

```
theta_path_bgd = []

def plot_gradient_descent(theta, eta, theta_path=None):
    m = len(X_b)
    plt.plot(X, y, "b.")
    n_iterations = 1000
    n_alpha = 0.1;
    for iteration in range(n_iterations):
        if iteration < 20:
            y_predict = X_new_b.dot(theta)
            if iteration > 0: # changed alpha setting to make direction of iterations clear
                style = "b-"
                plt.plot(X_new, y_predict, style, alpha=n_alpha)
                n_alpha = n_alpha + 0.01
            else:
                style = "r--"
                plt.plot(X_new, y_predict, style, alpha=1)

        gradients = 2/m * X_b.T.dot(X_b.dot(theta) - y)
        theta = theta - eta * gradients
        if theta_path is not None:
            theta_path.append(theta)
    plt.xlabel("$x_1$", fontsize=18)
    plt.axis([0, 2, 0, 15])
    plt.title(r"$\eta = {}$".format(eta), fontsize=16)
    plt.text(0.2, 13, r"$m = {0:.5f}$".format(float(theta[0])), fontsize=12)
    plt.text(0.2, 12, r"$c = {0:.5f}$".format(float(theta[1])), fontsize=12)

```

DIFFS

UCL SCHOOL OF MANAGEMENT

```
theta_path_bgd = []

def plot_gradient_descent(theta, eta, theta_path=None):
    m = len(X_b)
    plt.plot(X, y, "b.")
    n_iterations = 1000
    for iteration in range(n_iterations):
        if iteration < 10:
            y_predict = X_new_b.dot(theta)
            style = "b-" if iteration > 0 else "r--"
            plt.plot(X_new, y_predict, style)
        gradients = 2/m * X_b.T.dot(X_b.dot(theta) - y)
        theta = theta - eta * gradients
        if theta_path is not None:
            theta_path.append(theta)
    plt.xlabel("$x_1$", fontsize=18)
    plt.axis([0, 2, 0, 15])
    plt.title(r"$\eta = {}$".format(eta), fontsize=16)
```

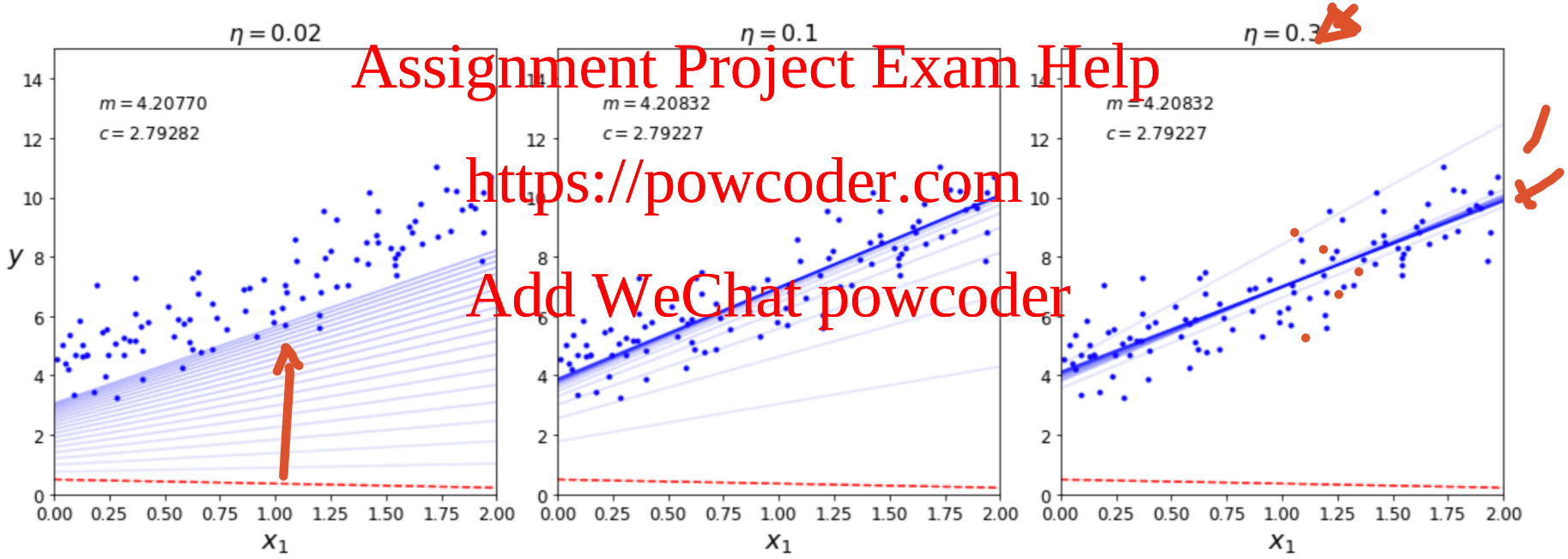
```
theta_path_bgd = []

def plot_gradient_descent(theta, eta, theta_path=None):
    m = len(X_b)
    plt.plot(X, y, "b.")
    n_iterations = 1000
    n_alpha = 0.1;
    for iteration in range(n_iterations):
        if iteration < 20:
            y_predict = X_new_b.dot(theta)
            if iteration > 0: # changed alpha setting to make direction of iterations clear
                style = "b-"
                plt.plot(X_new, y_predict, style, alpha=n_alpha)
                n_alpha = n_alpha + 0.01
            else:
                style = "r--"
                plt.plot(X_new, y_predict, style, alpha=1)

        gradients = 2/m * X_b.T.dot(X_b.dot(theta) - y)
        theta = theta - eta * gradients
        if theta_path is not None:
            theta_path.append(theta)
    plt.xlabel("$x_1$", fontsize=18)
    plt.axis([0, 2, 0, 15])
    plt.title(r"$\eta = {}$".format(eta), fontsize=16)
    plt.text(0.2, 13, r"$m = {0:.5f}$".format(float(theta[0])), fontsize=12)
    plt.text(0.2, 12, r"$c = {0:.5f}$".format(float(theta[1])), fontsize=12)
```

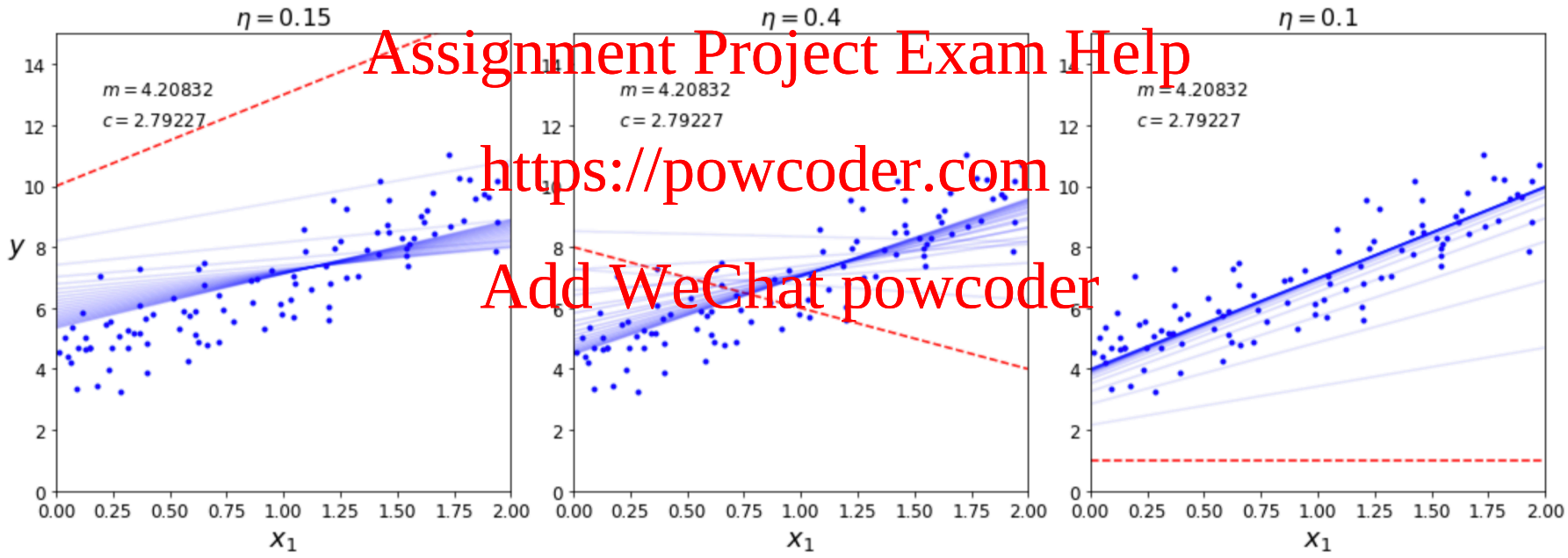
LEARNING RATES

UCL SCHOOL OF MANAGEMENT



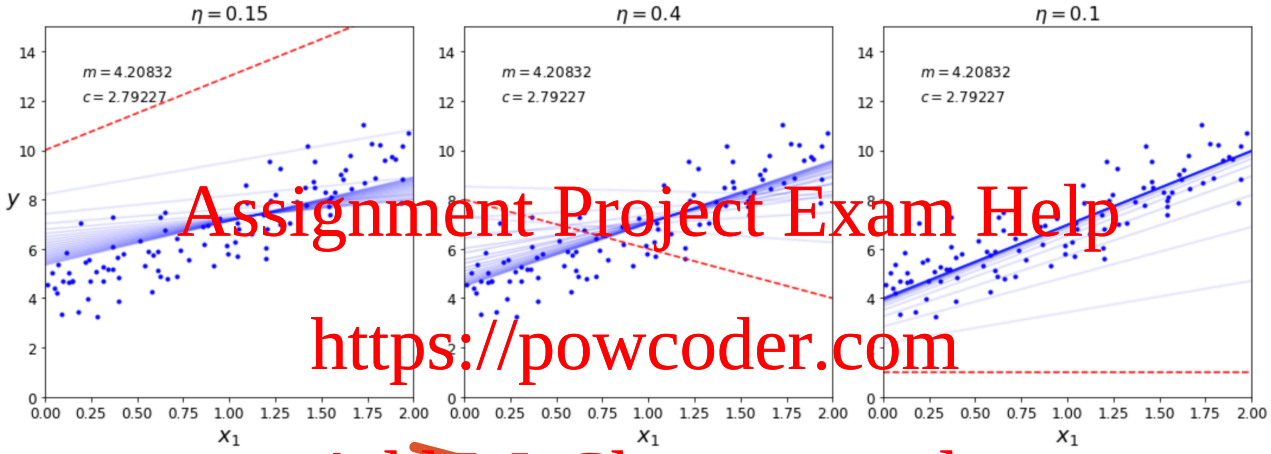
LEARNING RATES

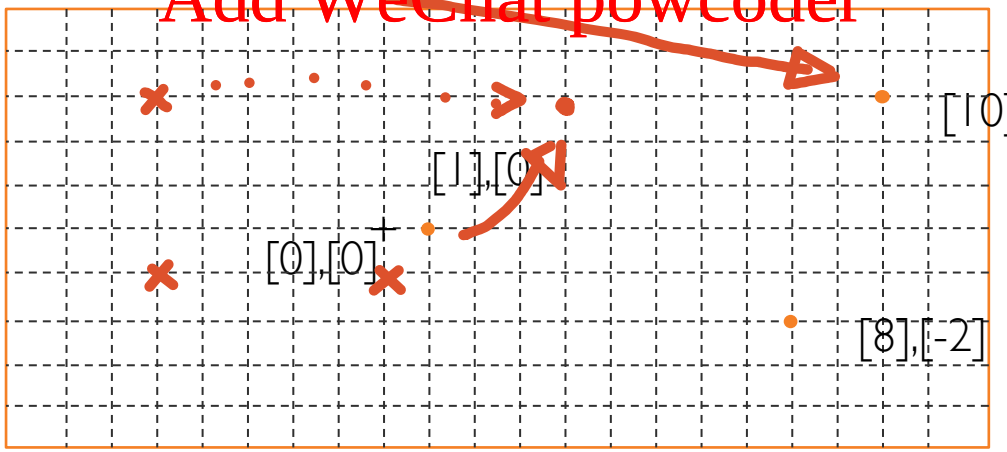
UCL
SCHOOL OF
MANAGEMENT



LEARNING RATES

UCL SCHOOL OF MANAGEMENT



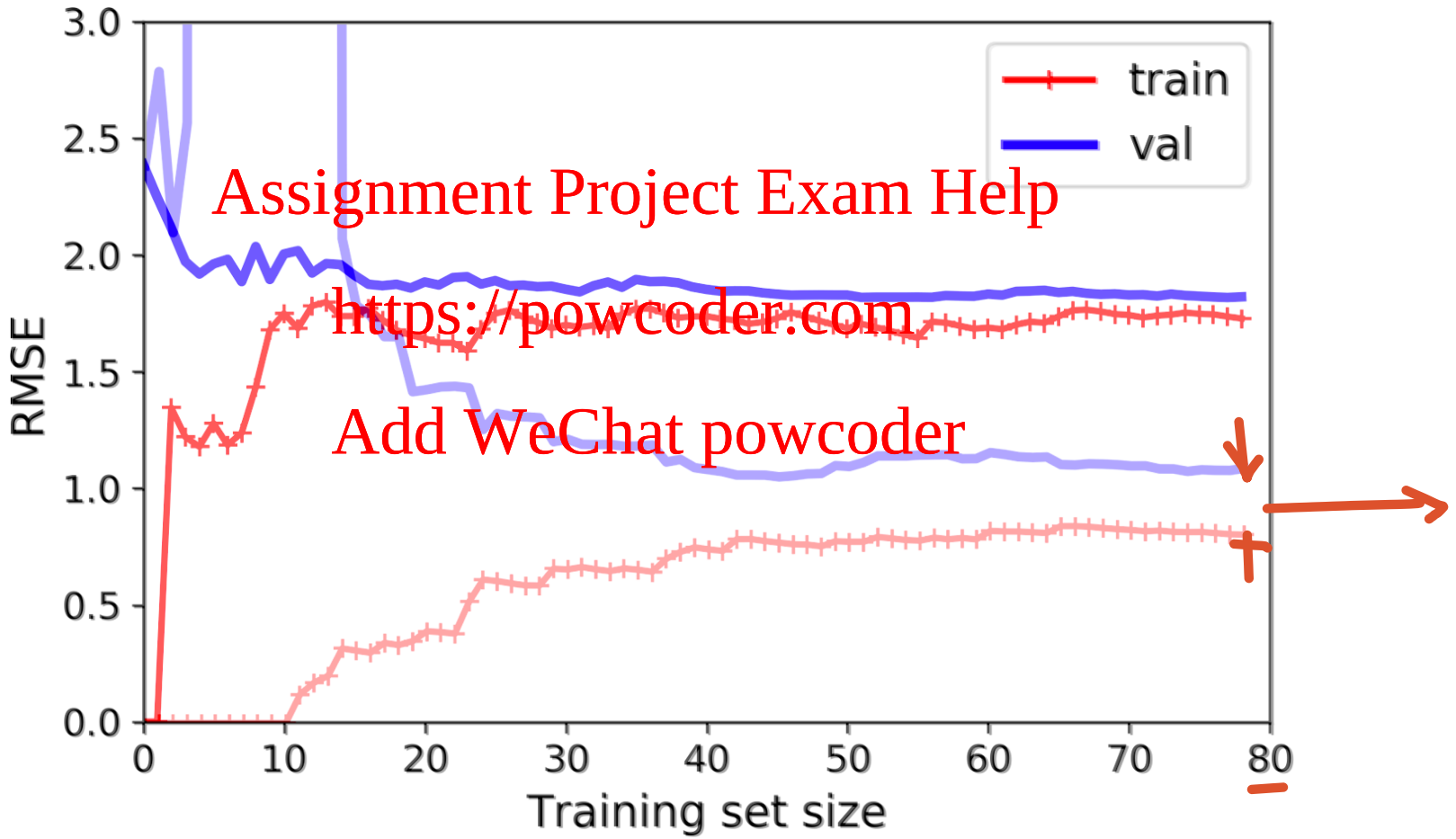


$\theta_1 = c$

$\theta_0 = m$

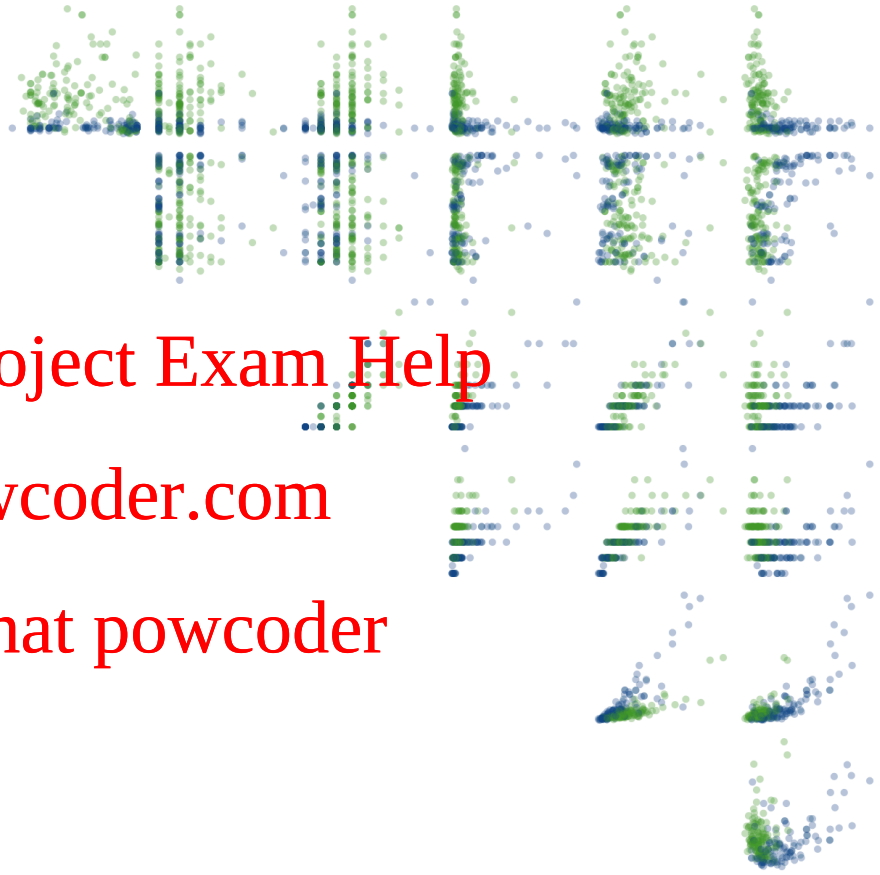
LEARNING CURVES

UCL SCHOOL OF MANAGEMENT



UCL
SCHOOL OF
MANAGEMENT

R2
D3



A visual introduction to machine learning

English

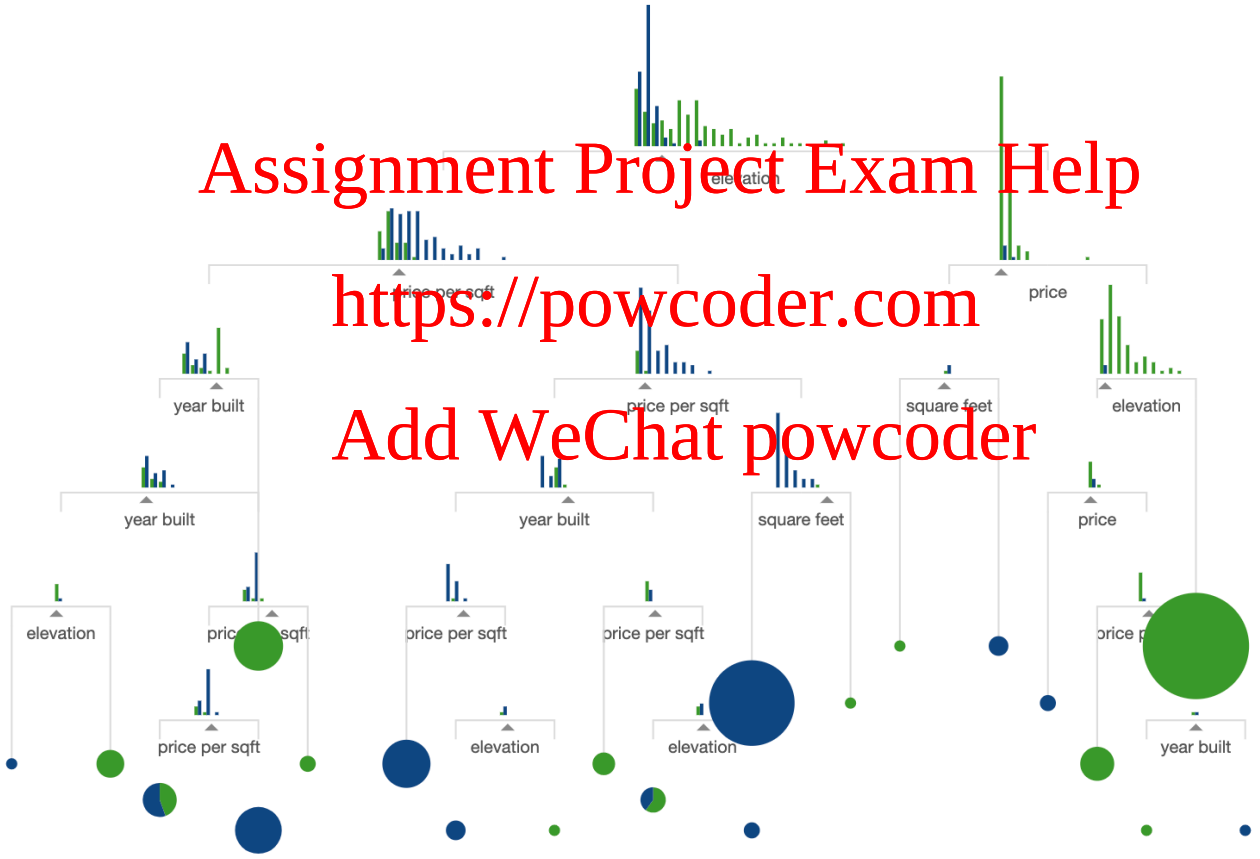
In machine learning, computers apply **statistical learning** techniques to automatically identify patterns in data. These techniques can be used to make highly accurate predictions.

Keep scrolling. Using a data set about homes, we will create a machine learning model to distinguish homes in New York from homes in San Francisco.

Source: http://www.r2d3.us/visual-intro-to-machine-learning-part-1/

TREE ALGORITHM

UCL
SCHOOL OF
MANAGEMENT



MODEL VARIANCE / 模型方差

MÓXÍNG FĀNGCHĀ

UCL SCHOOL OF MANAGEMENT

MODEL GENERALIZATION / 模型泛化

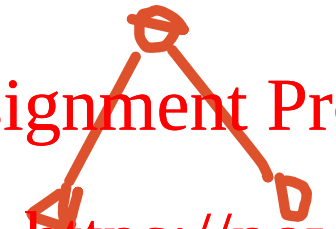
MÓXÍNG FÀN HUÀ

UCL SCHOOL OF MANAGEMENT

OVERFITTING / 过拟合

GUÒ NǏ HÉ

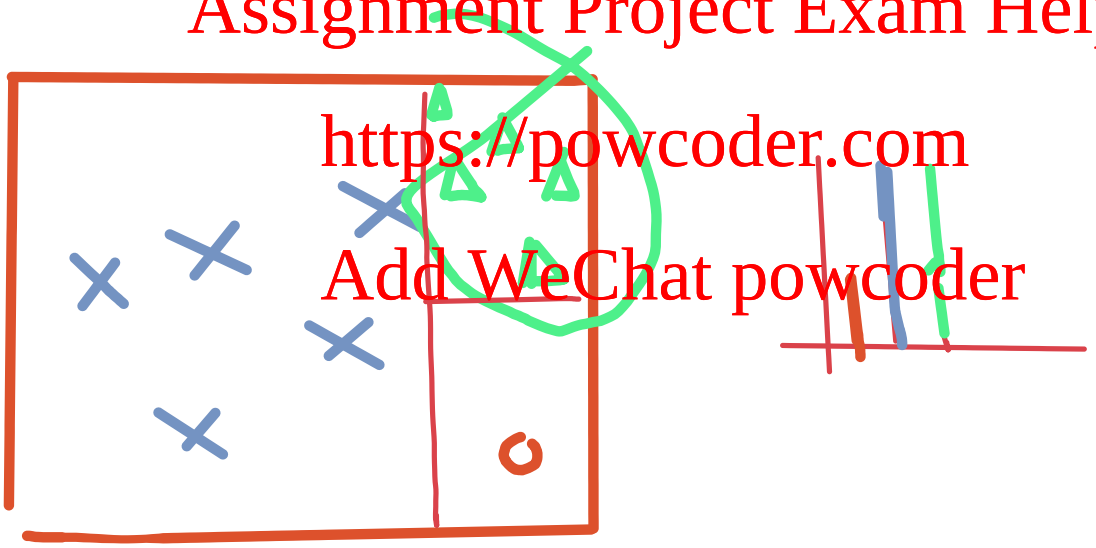
UCL
SCHOOL OF
MANAGEMENT



FAILURE MODES / 失败模式

SHĪBÀI MÓSHÌ

UCL SCHOOL OF MANAGEMENT



CLASS WEIGHTING

UCL
SCHOOL OF
MANAGEMENT

QUESTIONS

UCL SCHOOL OF MANAGEMENT

— What is the benefit of out-of-bag evaluation?

— If a Decision Tree is overfitting the training set, is it a good idea to try decreasing `max_depth`?

— If a Decision Tree is underfitting the training set, is it a good idea to try scaling the input features?

— What problems might we have if we try to grow a tree with high class imbalance?

Breakout 1

CLASS WEIGHTING

UCL SCHOOL OF MANAGEMENT

- **Domain expertise**
  - determined by talking to subject matter experts.
- **Tuning**
  - determined by a hyperparameter search such as a grid search.
- **Heuristic**
  - specified using a general best practice

Applications

- Fraud Detection
- Claim Prediction
- Churn Prediction
- Spam Detection

Breakout 2

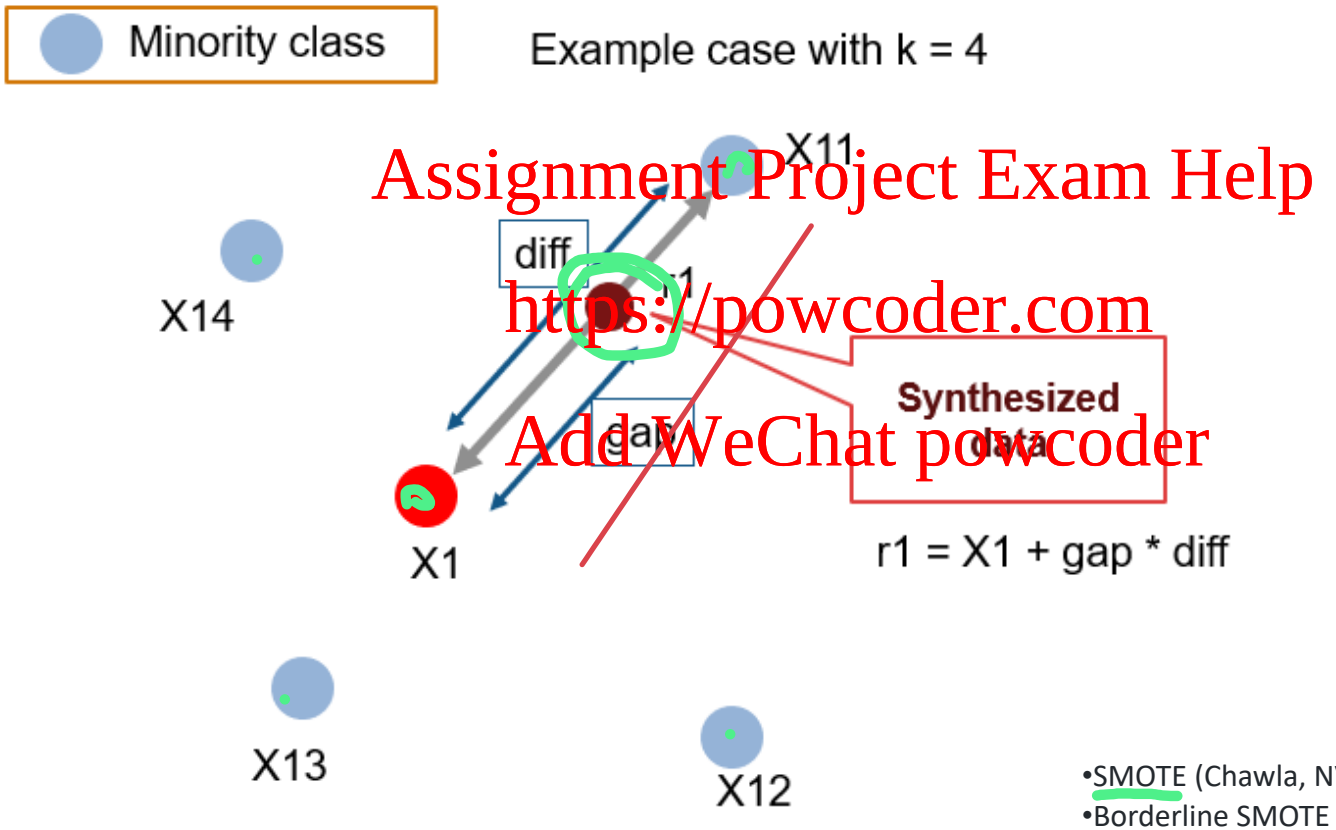
COMMON APPROACHES

UCL SCHOOL OF MANAGEMENT

- **Performance Metrics**
  - F-measure
  - G-mean
- **Data Sampling**
  - SMOTE (Synthetic Minority Oversampling Technique)
  - ENN (Edited nearest neighbours)
- **Cost-Sensitive Algorithms**
  - Decision Trees
- **Post-Processing**
  - Threshold Moving
  - Calibration

SYNTHETIC DATA SAMPLING

UCL SCHOOL OF MANAGEMENT



- SMOTE (Chawla, NV. et al. 2002)
- Borderline SMOTE (Han, H. et al. 2005)
- ADASYN (He, H. et al. 2008)
- Safe-level SMOTE (Bunkhumpornpat, C. at al. 2009)

Source: https://github.com/minoue-xx/Oversampling-Imbalanced-Data

Ensembles

KEY CONCEPTS

UCL
SCHOOL OF
MANAGEMENT

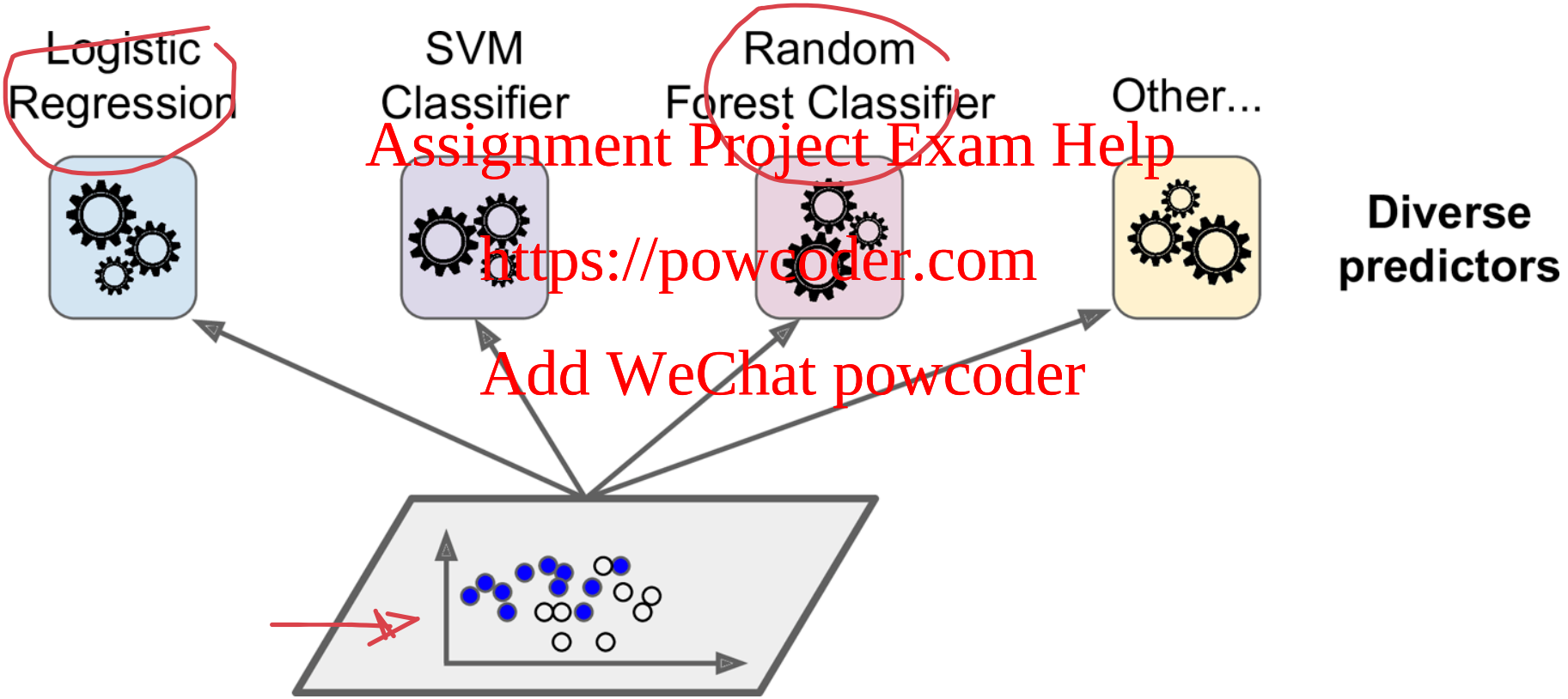
If you have trained five different models on the exact same training data, and they all achieve 95% precision, is there any chance that you can combine these models to get better results?

If so, how?

If not, why?

MULTIPLE MODELS

UCL
SCHOOL OF
MANAGEMENT



VOTING

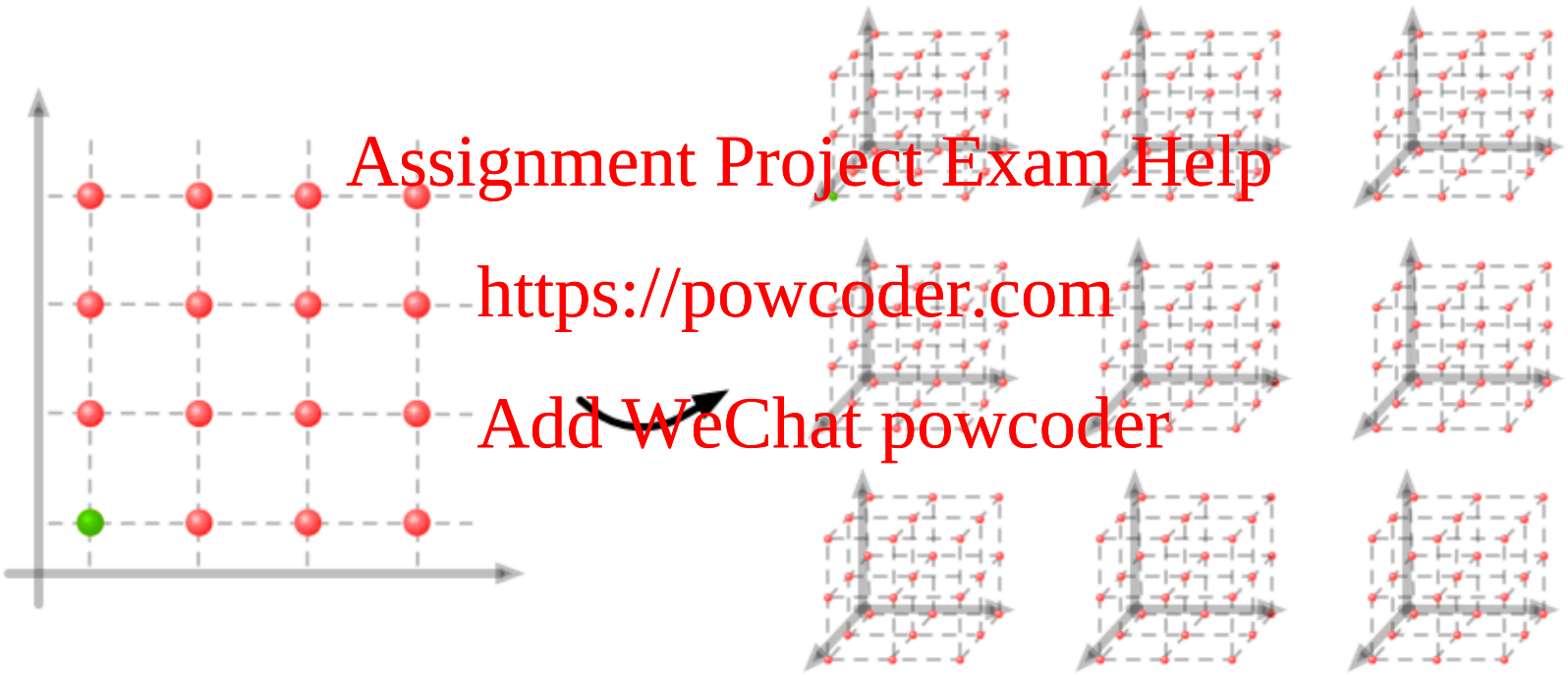
UCL
SCHOOL OF
MANAGEMENT

Leipzig–Dresden Railway Company in 1852

Why is ML hard?

WHY IS ML HARD?

UCL
SCHOOL OF
MANAGEMENT



Software Development

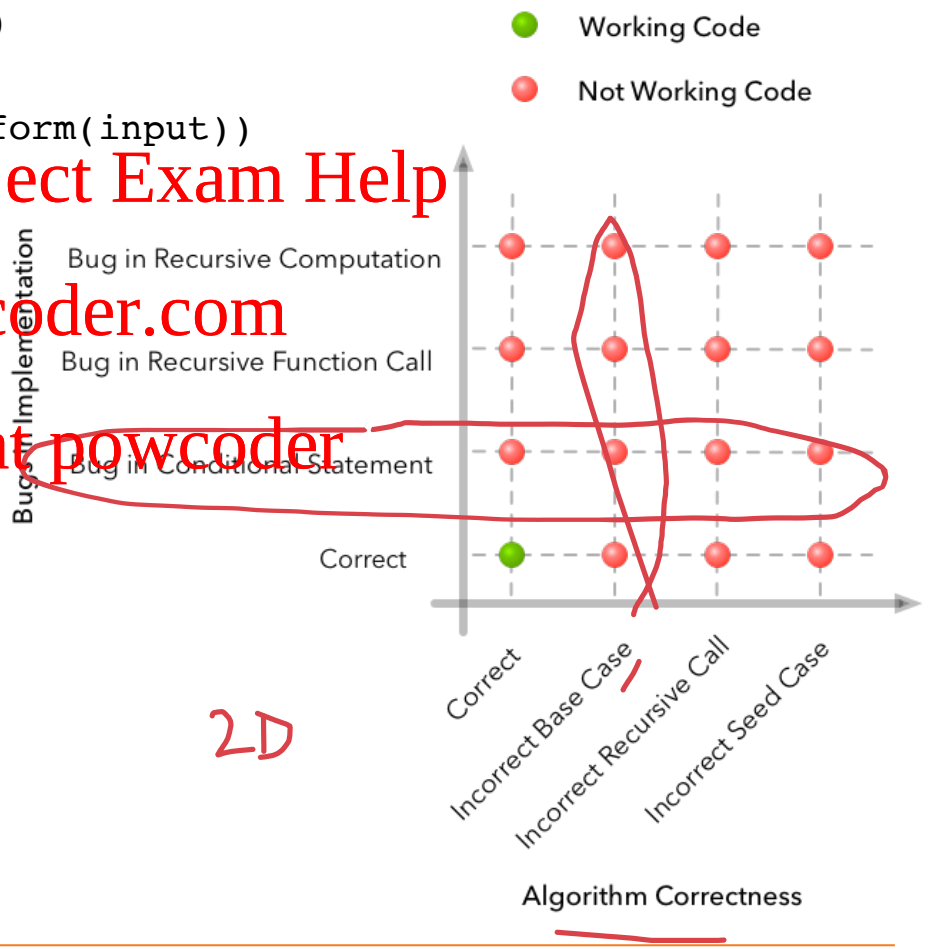
Machine Learning

https://ai.stanford.edu/~zayd/why-is-machine-learning-hard.html

DEBUGGING

UCL SCHOOL OF MANAGEMENT

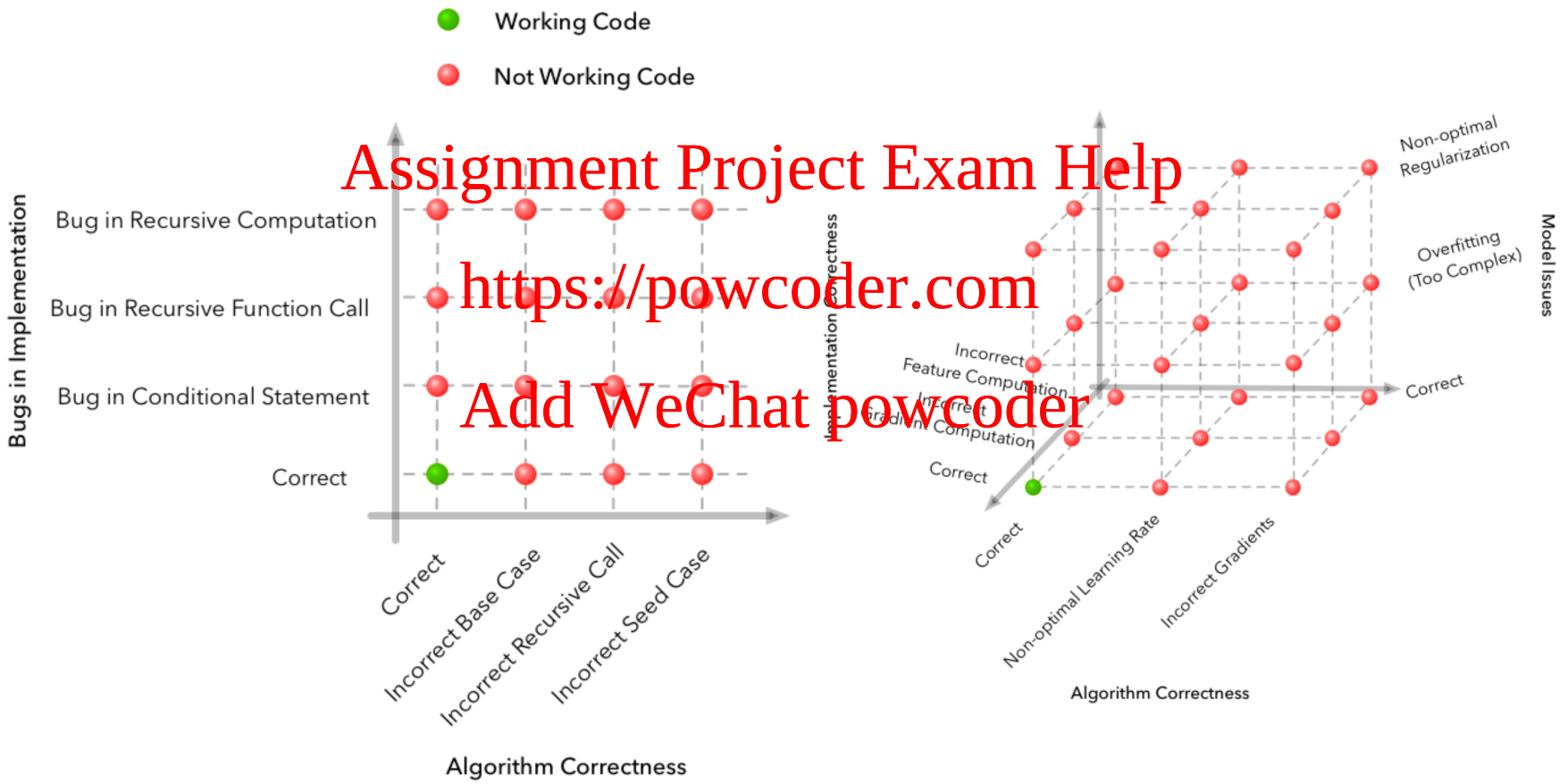
```
def recursion(input):
        if input is endCase:
                return transform(input)
        else:
                return recursion(transform(input))
```

WHY IS ML HARD?

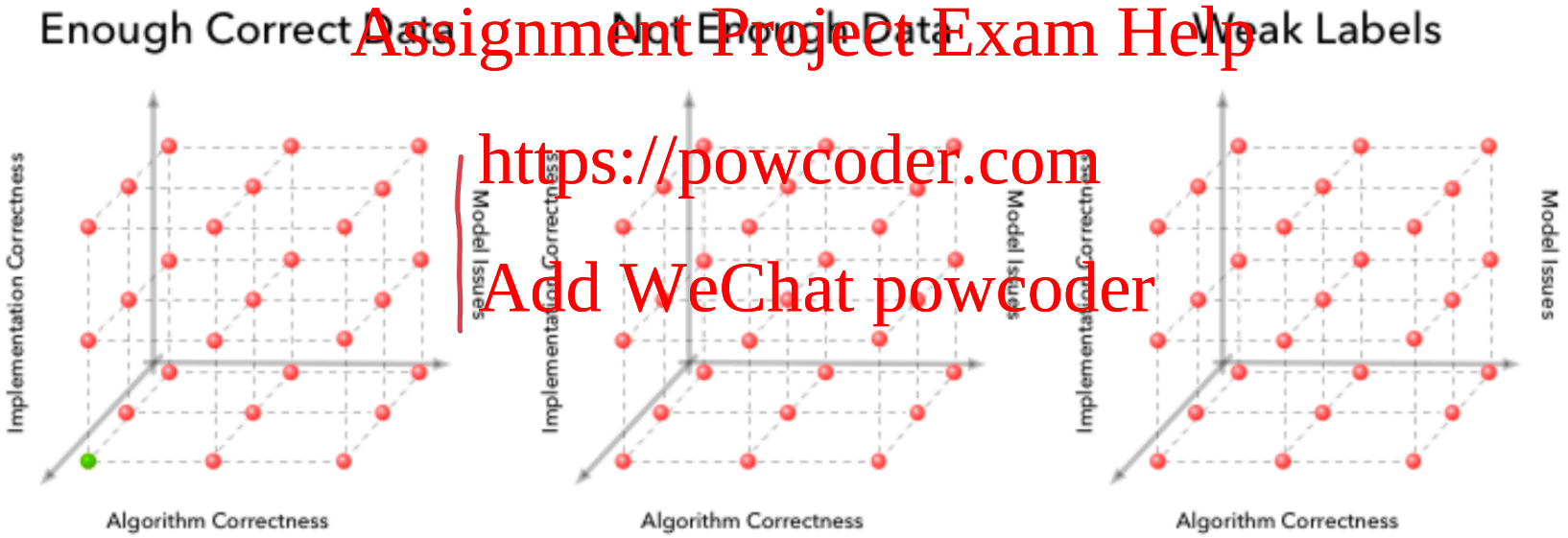
ALGORITHM, IMPLEMENTATION, DATA, MODEL

UCL SCHOOL OF MANAGEMENT



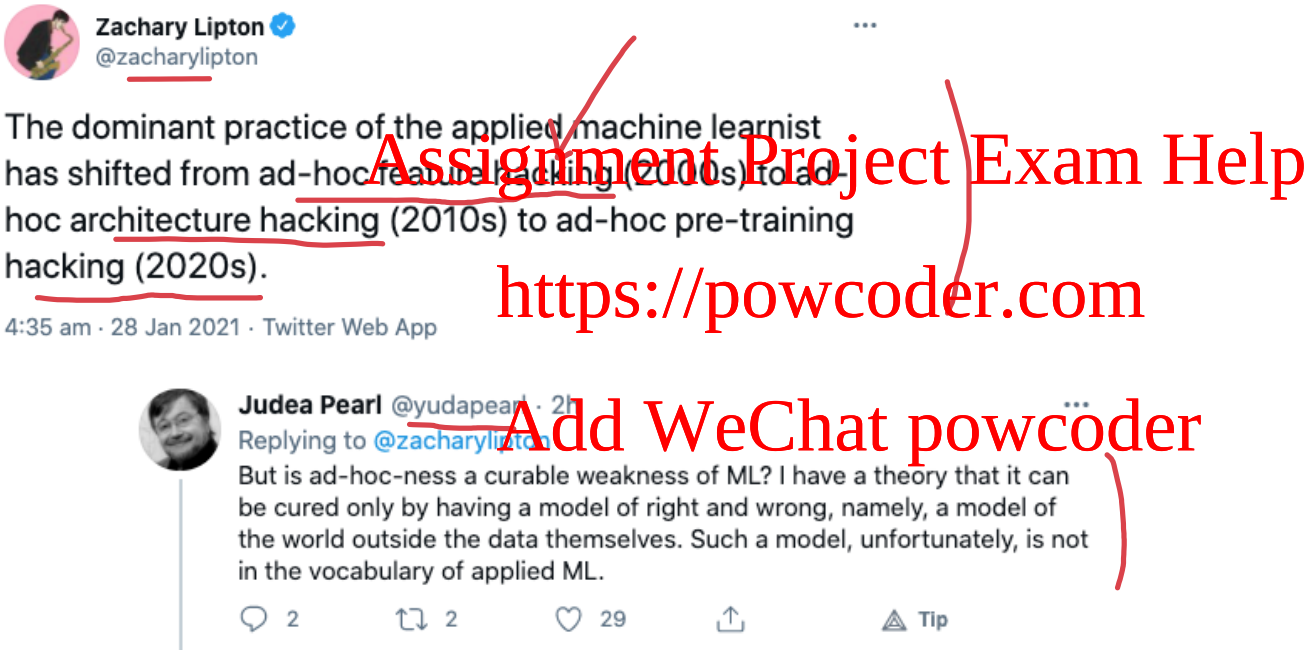
WHY IS ML HARD?

UCL SCHOOL OF MANAGEMENT



ML AS HACKING

UCL SCHOOL OF MANAGEMENT



Zachary Lipton
@zacharylipton

The dominant practice of the applied machine learnist has shifted from ad-hoc feature hacking (2000s) to ad-hoc architecture hacking (2010s) to ad-hoc pre-training hacking (2020s).

4:35 am · 28 Jan 2021 · Twitter Web App

Judea Pearl @yudapearl · 2h
Replying to @zacharylipton
But is ad-hoc-ness a curable weakness of ML? I have a theory that it can be cured only by having a model of right and wrong, namely, a model of the world outside the data themselves. Such a model, unfortunately, is not in the vocabulary of applied ML.

2 2 29 Tip

TWITTER

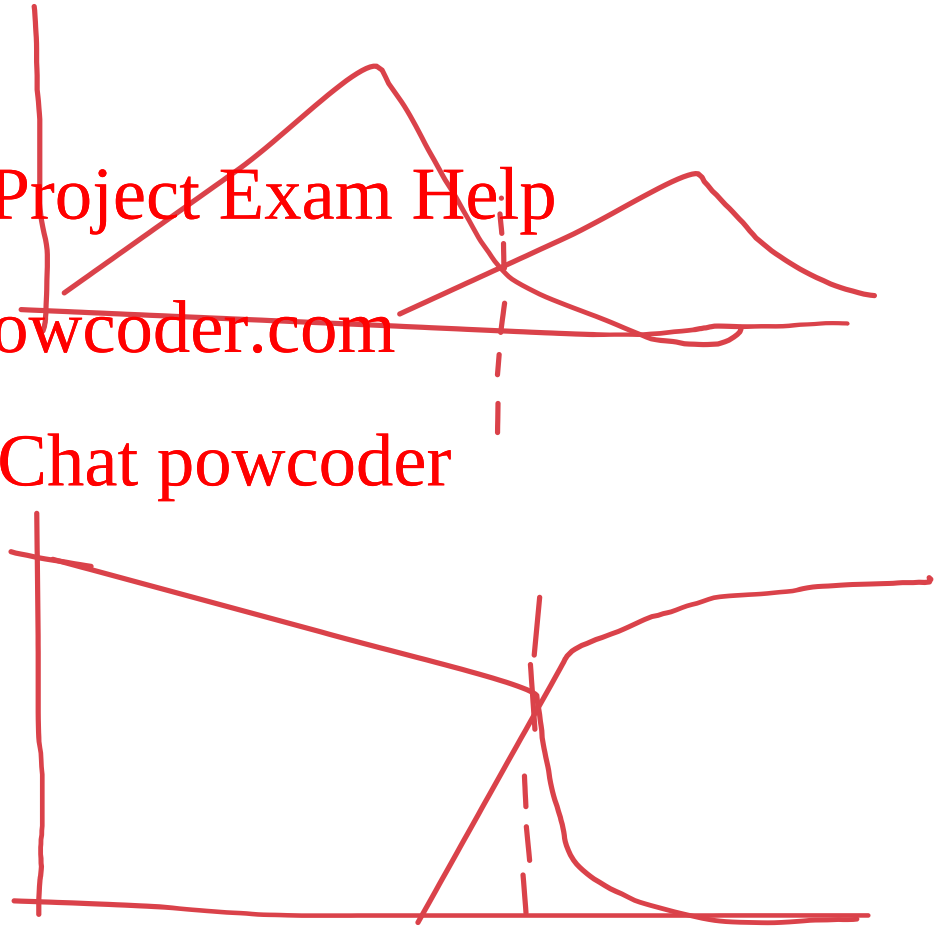
UCL
SCHOOL OF
MANAGEMENT

@chipro

@random_forests

@zacharylipton

@yudapearl



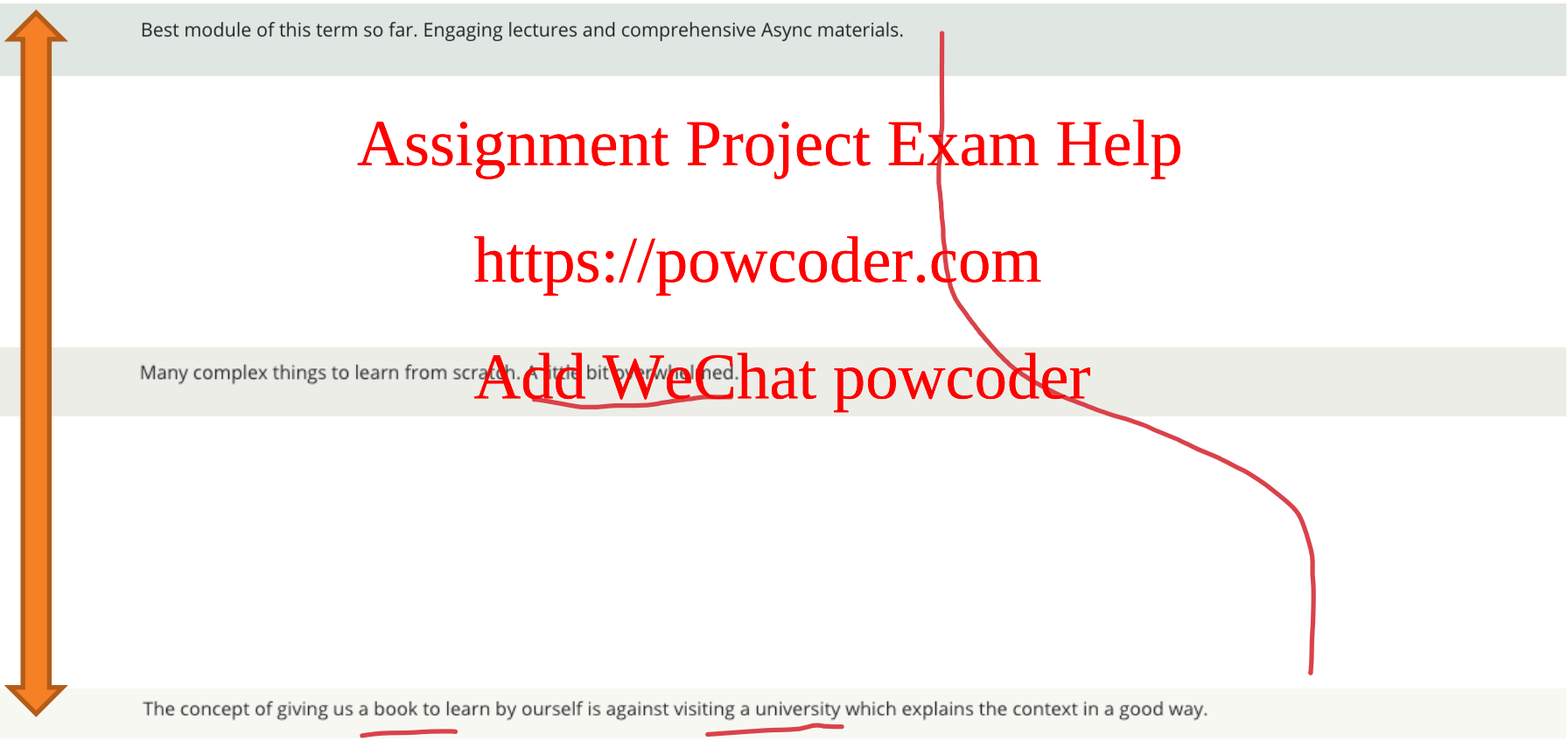
HAVE YOUR SAY!

UCL SCHOOL OF MANAGEMENT

"The questionnaires are very short and will take less than a minute for them to complete."

VOTING IN ACTION!

UCL SCHOOL OF MANAGEMENT



TEACHING TEAM

UCL
SCHOOL OF
MANAGEMENT

Dr Alastair Moore
Senior Teaching Fellow
a.p.moore@ucl.ac.uk
@latticecut

Kamil Tylinski
Teaching Assistant
kamil.tylinski.16@ucl.ac.uk

Jiangbo Shangguan
Teaching Assistant
j.shangguan.17@ucl.ac.uk

LECTURE 3 TERM 2:

MSIN0097

UCL SCHOOL OF MANAGEMENT

PREDICTIVE ANALYTICS

A P MOORE