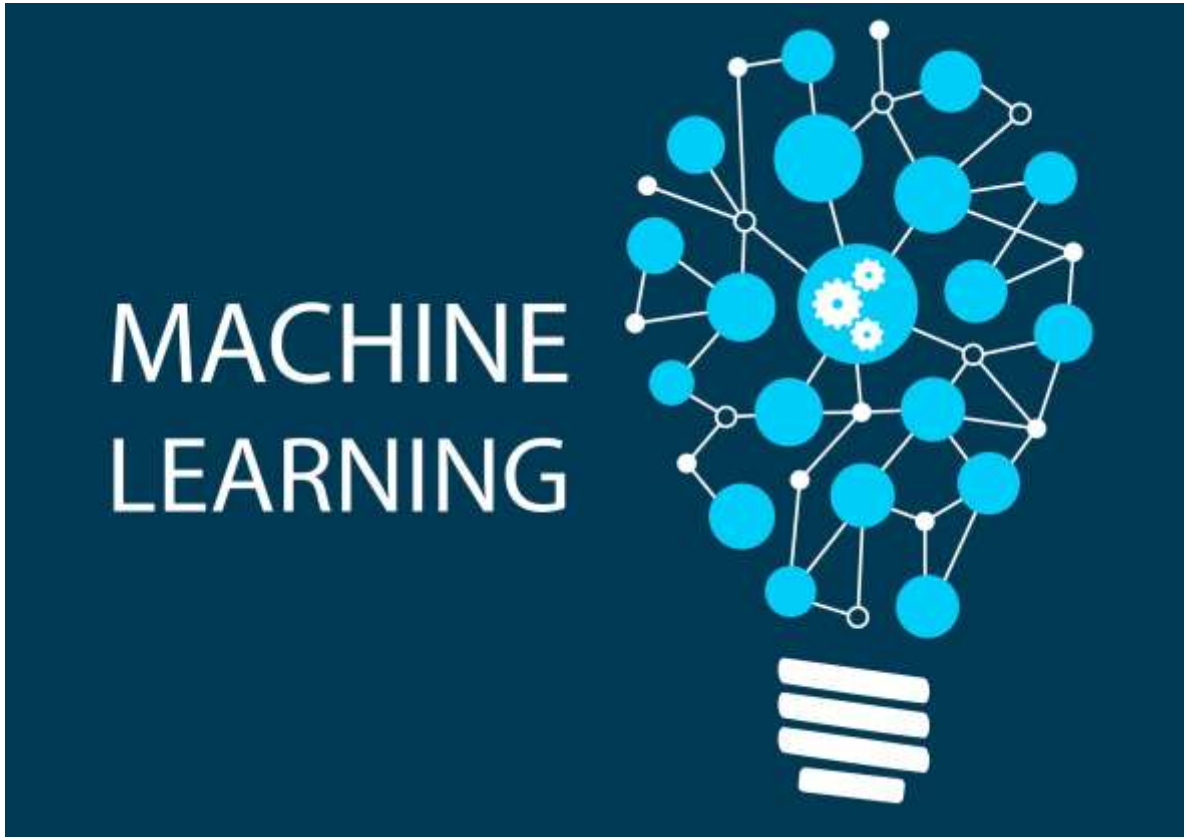


Multiple Linear Regression

Equation of simple linear regression

$$\hat{y} = \theta_0 + \theta_1 x_1$$

Y= m1.x1+b

m1 → coefficient

x1 → independent variable

b → intercept

	area	Price
0	2600	550000
1	3000	565000
2	3200	610000
3	3600	680000
4	4000	725000

- Price= m1*area + b

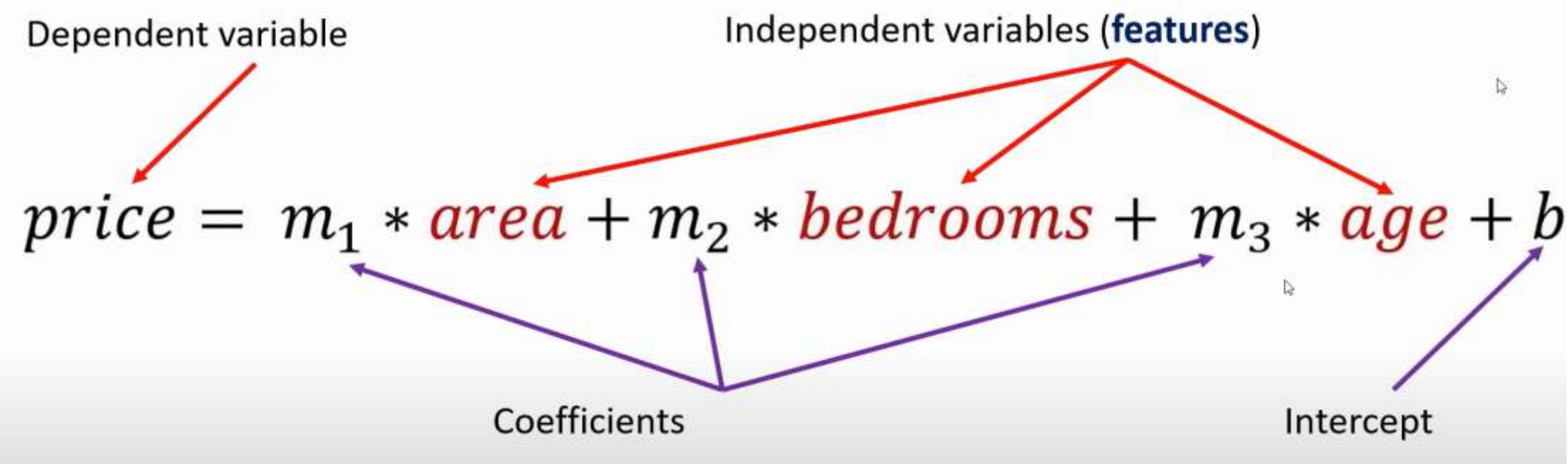
Multiple Linear Regression

$$price = m_1 * area + m_2 * bedrooms + m_3 * age + b$$

Simple Linear regression

Price= m1*area + b

Details...



Price= m1*area+b → simple Linear regression

Generally…

$$y = m_1 x_1 + m_2 x_2 + m_3 x_3 + b$$

Topics

- Data Preprocessing: Handling NA values
- Linear Regression Using Multiple Variables

Exemple

Home prices in Monroe Township, NJ (USA)

area	bedrooms	age	price
2600	3	20	550000
3000	4	15	565000
3200		18	610000
3600	3	30	595000
4000	5	8	760000

Given these home prices find out price of a home that has,

3000 sqr ft area, 3 bedrooms, 40 year old
2500 sqr ft area, 4 bedrooms, 5 year old

Home work

1	experience	test_score(out of 10)	interview_score(out of 10)	salary($)
2		8	9	50000
3		8	6	45000
4	five	6	7	60000
5	two	10	10	65000
6	seven	9	6	70000
7	three	7	10	62000
8	ten		7	72000
9	eleven	7	8	80000
10				

Lecture 4

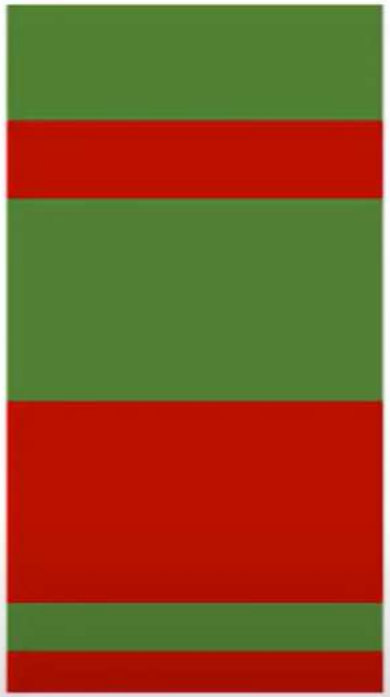
Model Evaluation Approaches

Le Train-Test Split

Avec le Train-Test Split, on commence par décomposer l'ensemble de données de façon aléatoire. Une partie permettra d'entraîner le modèle de Machine Learning, tandis que l'autre servira pour le test de validation. Dans la majorité des cas, **70 à 80% des données du dataset sont utilisées pour l'entraînement**. Le reste sera exploité dans le cadre de la Cross-Validation.

1. Use all available data for training and testing and test on the same dataset

2. Split available dataset into training & test sets



70 math questions

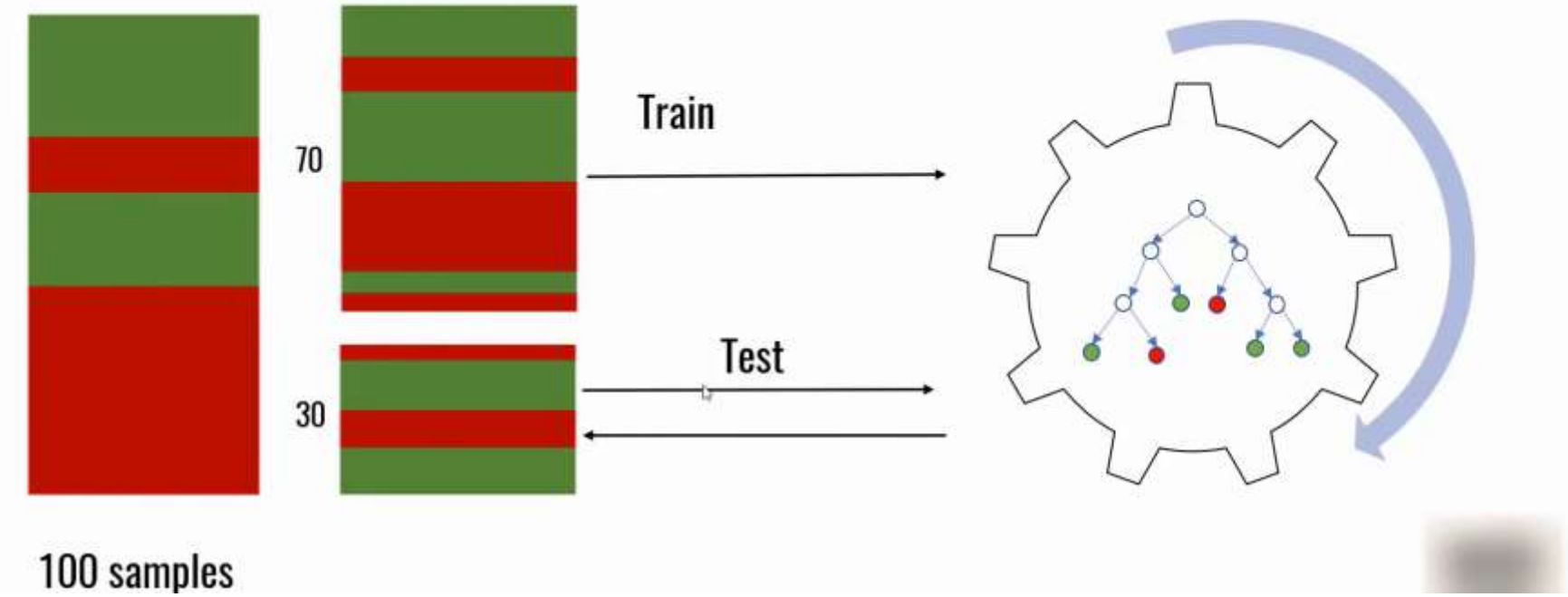
Train

Test

Remaining 30 questions

Le train_test_split Model

```
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X,y,test_size=0.3)
```



Train/test split Evaluation

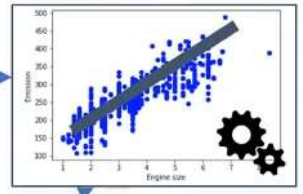
Dans cette approche, nous sélectionnons une partie de notre ensemble de données pour l'entraînement, par exemple, les lignes zéro à cinq, et le reste est utilisé pour tester, par exemple, les lignes six à neuf. Le modèle est construit sur l'ensemble d'apprentissage. Ensuite, l'ensemble de fonctionnalités de test est transmis au modèle pour la prédiction. Enfin, **les valeurs prédites pour l'ensemble de test** sont **comparées** aux **valeurs réelles de l'ensemble de test**. Après quoi, vous vous entraînez avec l'ensemble d'entraînement et testez avec l'ensemble de test. Cela fournira une évaluation plus précise de la précision hors échantillon**, car l'ensemble de données de test ne fait pas partie de l'ensemble de données qui a été utilisé pour entraîner les données**. C'est plus réaliste pour les problèmes du monde réel.

Train/Test Split evaluation

	ENGINESIZE	CYLINDERS	FUELCONSUMPTION_COMB	CO2EMISSIONS
0	2.0	4	8.5	196
1	2.4	4	9.6	221
2	1.5	4	5.9	136
3	3.5	6	11.1	255
4	3.5	6	10.6	244
5	3.5	6	10.0	230
6	3.5	6	10.1	232
7	3.7	6	11.1	255
8	3.7	6	11.6	267
9	2.4	4	9.2	212

Train

Test



	Prediction
6	234
7	256
8	267
9	210

Actual values

Predicted values

Train/test split Evaluation

In this approach, we select a portion of our dataset for training, for example, row zero to five,and the rest is used for testing, for example, row six to nine.

The model is built on the training set.Then, the test feature set is passed to the model for prediction. Finally, the predicted values for the test set are compared with the actual values of the testing set. The second evaluation approach is called train/test split.Train/test split involves splitting the dataset into training and testing sets respectively, which are mutually exclusive. After which, you train with the **training set** and **test with the testing set**. This will provide a more accurate evaluation on out-of-sample accuracy because the testing dataset is not part of the dataset that has been used to train the data. It is more realistic for real-world problems.

Remarque/Remark

Étant donné que ces données n'ont pas été utilisées pour entraîner le modèle, le modèle n'a aucune connaissance du résultat de ces points de données. Il s'agit donc essentiellement de tests hors échantillon. Cependant, assurez-vous d'entraîner votre modèle avec l'ensemble de test par la suite, car vous ne voulez pas perdre de données potentiellement précieuses.

Since this data has not been used to train the model,the model has no knowledge of the outcome of these data points.So, in essence, it's truly out-of-sample testing. However, please ensure that you train your model with the testing set afterwards ,as you don't want to lose potentially valuable data.

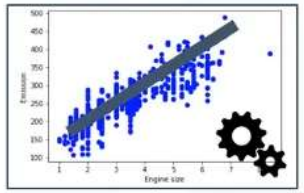
Best model for most accurate results

	ENGINESIZE	CYLINDERS	FUELCONSUMPTION_COMB	CO2EMISSIONS
0	2.0	4	8.5	196
1	2.4	4	9.6	221
2	1.5	4	5.9	136
3	3.5	6	11.1	255
4	3.5	6	10.6	244
5	3.5	6	10.0	230
6	3.5	6	10.1	232
7	3.7	6	11.1	255
8	3.7	6	11.6	267
9	2.4	4	9.2	212

Train

Test

Actual values



	Prediction
6	234
7	256
8	267
9	210

Predicted values

Remarque/Remark

Maintenant, nous passons l'ensemble de fonctionnalités de la partie de test à notre modèle construit et prédisons les valeurs cibles. Enfin, nous comparons les valeurs prédites par notre modèle avec les valeurs réelles de l'ensemble de test. Cela indique à quel point notre modèle est précis.

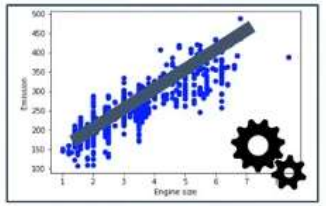
Now we pass the feature set of the testing portion to our built model and predict the target values.Finally, we compare the predicted values by our model with the actual values in the test set. This indicates how accurate our model actually is.

Resultat/Result

	ENGINESIZE	CYLINDERS	FUELCONSUMPTION_COMB	CO2EMISSIONS
0	2.0	4	8.5	196
1	2.4	4	9.6	221
2	1.5	4	5.9	136
3	3.5	6	11.1	255
4	3.5	6	10.6	244
5	3.5	6	10.0	230
6	3.5	6	10.1	232
7	3.7	6	11.1	255
8	3.7	6	11.6	267
9	2.4	4	9.2	212

Train

Test



	Prediction
6	234
7	256
8	267
9	210

Actual values

Predicted values

Exemple: BMW dataset

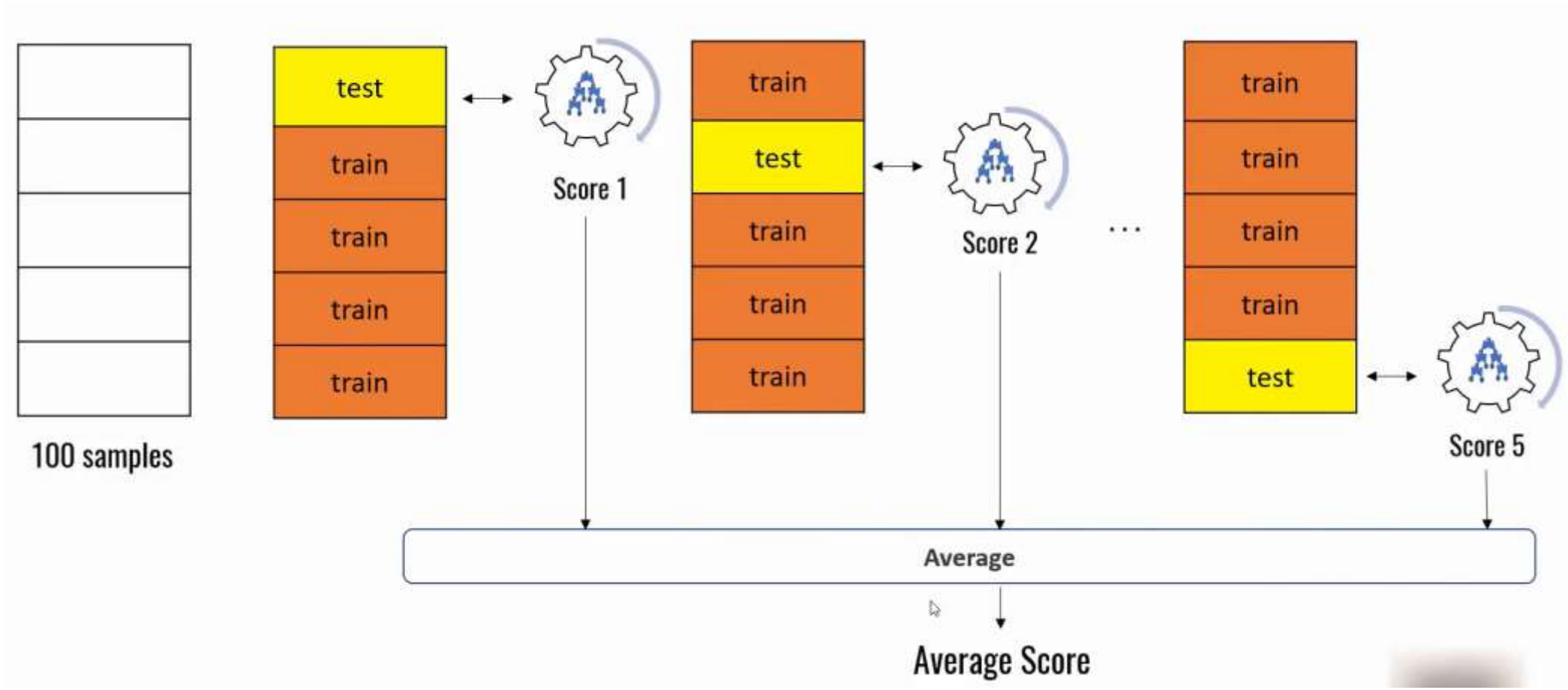
Mileage	Age(yrs)	Sell Price($)
69000	6	18000
35000	3	34000
57000	5	26100
22500	2	40000
46000	4	31500
59000	5	26750
52000	5	32000
72000	6	19300
91000	8	12000
67000	6	22000

Slip_train_test

	Mileage	Age(yrs)	Sell Price($)
Training (80%)	69000	6	18000
	35000	3	34000
	57000	5	26100
	22500	2	40000
	46000	4	31500
	59000	5	26750
	52000	5	32000
	72000	6	19300
Test	91000	8	12000
	67000	6	22000

Problem with techniques…

Cross validation techniques



Questions?

Calculating the accuracy of the Model
Calcul de la précision du modèle

Maintenant, nous passons l'ensemble de fonctionnalités de la partie de test à notre modèle construit et prédisons les valeurs cibles. Enfin, nous comparons les valeurs prédites par notre modèle avec les valeurs réelles de l'ensemble de test. Cela indique à quel point notre modèle est précis.

Now we pass the feature set of the testing portion to our built model and predict the target values. Finally, we compare the predicted values by our model with the actual values in the test set. This indicates how accurate our model actually is.

Calcul de la précision du modèle

	ENGINESIZE	CYLINDERS	FUELCONSUMPTION_COMB	CO2EMISSIONS
0	2.0	4	8.5	196
1	2.4	4	9.6	221
2	1.5	4	5.9	136
3	3.5	6	11.1	255
4	3.5	6	10.6	244
5	3.5	6	10.0	230
6	3.5	6	10.1	232
7	3.7	6	11.1	255
8	3.7	6	11.6	267
9	2.4	4	9.2	212

Test

y

Actual values

$$Error = \frac{(232 - 234) + (255 - 256) + \ \ldots}{4}$$

$$Error = \frac{1}{n}\sum_{j=1}^{n}|y_j - \hat{y}_j|$$

$\hat{y}$

	Prediction
6	234
7	256
8	267
9	210

Predicted values

when you test with a dataset in which you know the target value for each data point, you're able to obtain a percentage of accurate predictions for the model. This evaluation approach would most likely have a **high training accuracy** and the low **out-of-sample accuracy** since the model knows all of the testing data points from the training.

What is training accuracy and out-of-sample accuracy? We said that training and testing on the same dataset produces a high training accuracy, but what exactly is training accuracy? Training accuracy is the percentage of correct predictions that the model makes when using the test dataset. However, a high training accuracy isn't necessarily a good thing. For instance, having a high training accuracy may result in an over-fit the data. This means that the model is overly trained to the dataset, which may capture noise and produce a non-generalized model. **Out-of-sample accuracy is the percentage of correct predictions** that the model makes on data that the model has not been trained on.

lorsque vous testez avec un ensemble de données dans lequel vous connaissez la valeur cible pour chaque point de données, vous êtes en mesure d'obtenir un pourcentage de prédictions précises pour le modèle.

Cette approche d'évaluation aurait très probablement une grande précision d'entraînement et la faible précision hors échantillon depuis le modèle connaît tous les points de données de test de la formation. **Qu'est-ce que la précision de l'entraînement** et **la précision hors échantillon ?** Nous avons dit que l'entraînement et les tests sur le même ensemble de données produisent une précision d'entraînement élevée, mais qu'est-ce que la précision de l'entraînement ? La précision de l'entraînement est le pourcentage de les prédictions correctes que le modèle fait lors de l'utilisation de l'ensemble de données de test. Cependant, une précision d'entraînement élevée n'est pas nécessairement une bonne chose. Par exemple, une précision d'entraînement élevée peut entraîner un surajustement des données. Cela signifie que le modèle est trop entraîné à l'ensemble de données, qui peut capturer le bruit et produire un modèle non généralisé. La précision hors échantillon est le pourcentage de prédictions correctes qui le modèle utilise des données sur lesquelles le modèle n'a pas été formé.

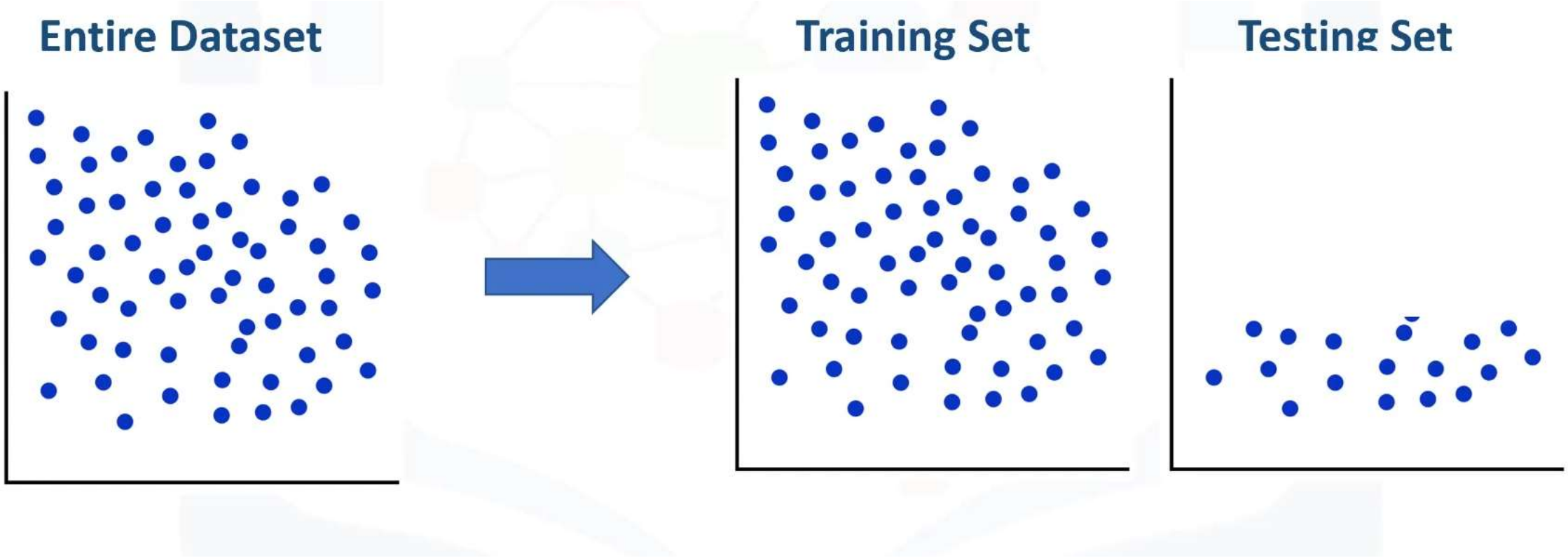
So, how can we improve out-of-sample accuracy? One way is to use another evaluation approach called train/test split. In this approach, we select a portion of our dataset for training, for example, row zero to five, and the rest is used for testing, for example, row six to nine. The model is built on the training set. Then, the test feature set is passed to the model for prediction. Finally, the predicted values for the test set are compared with the actual values of the testing set. The second evaluation approach is called train/test split. Train/test split involves splitting the dataset into training and testing sets respectively, which are mutually exclusive.

Alors, comment pouvons-nous améliorer la précision hors échantillon ? Une façon consiste à utiliser une autre approche d'évaluation appelée fractionnement train/test. Dans cette approche, nous sélectionnons une partie de notre ensemble de données pour **la formation(training set)**, par exemple, ligne zéro à cinq, et le reste est utilisé pour **les tests(testing set),** par exemple, rangée six à neuf. Le modèle est construit sur l'ensemble d'entraînement(training set). Ensuite, l'ensemble de fonctionnalités de test est transmis au modèle pour la prédiction. Enfin, les valeurs prédites pour l'ensemble de test sont comparés aux valeurs réelles de l'ensemble de test(Actual values of the testing set). La deuxième approche d'évaluation est appelée fractionnement train/test(**train/test split**). La division Train/Test implique la division de l'ensemble de données respectivement en ensembles d'apprentissage et de test, qui s'excluent mutuellement

Cela fournira une évaluation plus précise de la précision hors échantillon, car l'ensemble de données de test ne fait pas partie de l'ensemble de données qui a été utilisé pour entraîner les données. C'est plus réaliste pour les problèmes du monde réel.

The second evaluation approach is called train/test split. Train/test split involves splitting the dataset into training and testing sets respectively, which are mutually exclusive. After which, you train with the training set and test with the testing set. This will provide a more accurate evaluation on out-of-sample accuracy because the testing dataset is not part of the dataset that has been used to train the data. It is more realistic for real-world problems.

Train and test on the same dataset
Entrainer et tester sur le même dataset



What is training and out of sample accuracy?

- **Training Accuracy**
  - High training accuracy isn't necessarily a good thing
  - Result of over-fitting
    - Over-fit: the model is overly trained to the dataset, which may capture noise and produce a non-generalized model
- **Out-of-Sample Accuracy**
  - It's important that our models have a high, out-of-sample accuracy
  - How can we improve out-of-sample accuracy?

Remark/Remarque

The issue with train/test split is that it's highly dependent on the datasets on which the data was trained and tested. The variation of this causes train/test split to have a better out-of-sample prediction than training and testing on the same dataset, but it still has some problems due to this dependency.

Le problème avec la division train/test est qu'elle dépend fortement des ensembles de données sur lesquels les données ont été formées et testées. La variation de ceci fait que la division train/test a une meilleure prédiction hors échantillon que la formation et le test sur le même ensemble de données, mais il a encore quelques problèmes en raison de cette dépendance.

How to use k-fold cross validation?

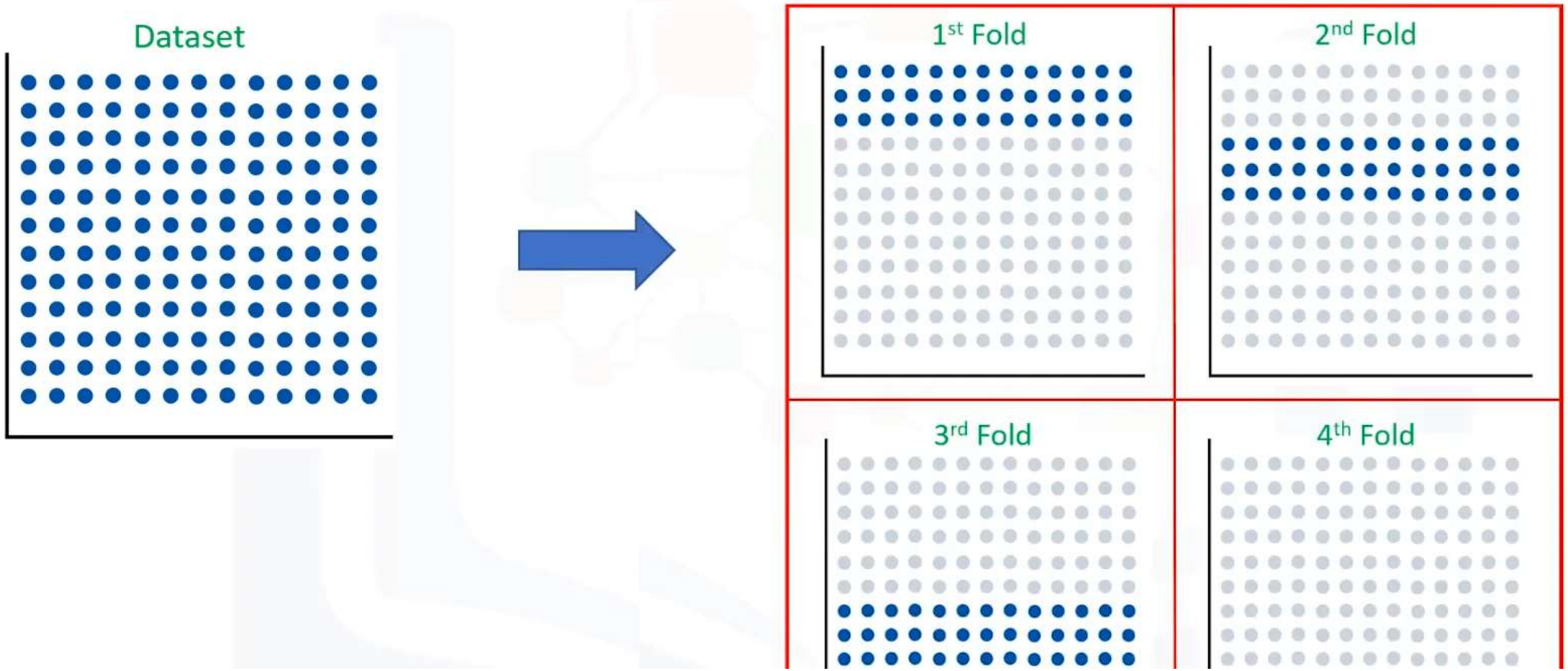
The entire dataset is represented by the points in the image at the top left. If we have K equals four folds, then we split up this dataset as shown here. In the first fold for example, we use the first 25 percent of the dataset for testing and the rest for training. The model is built using the training set and is evaluated using the test set. Then, in the next round or in the second fold, the second 25 percent of the dataset is used for testing and the rest for training the model. Again, the accuracy of the model is calculated. We continue for all folds. Finally, the result of all four evaluations are averaged. That is, the accuracy of each fold is then averaged, keeping in mind that each fold is distinct, where no training data in one fold is used in another.

K-fold cross-validation in its simplest form performs multiple train/test splits, using the same dataset where each split is different. Then, the result is average to produce a more consistent out-of-sample accuracy. We wanted to show you an evaluation model that addressed some of the issues we've described in the previous approaches.

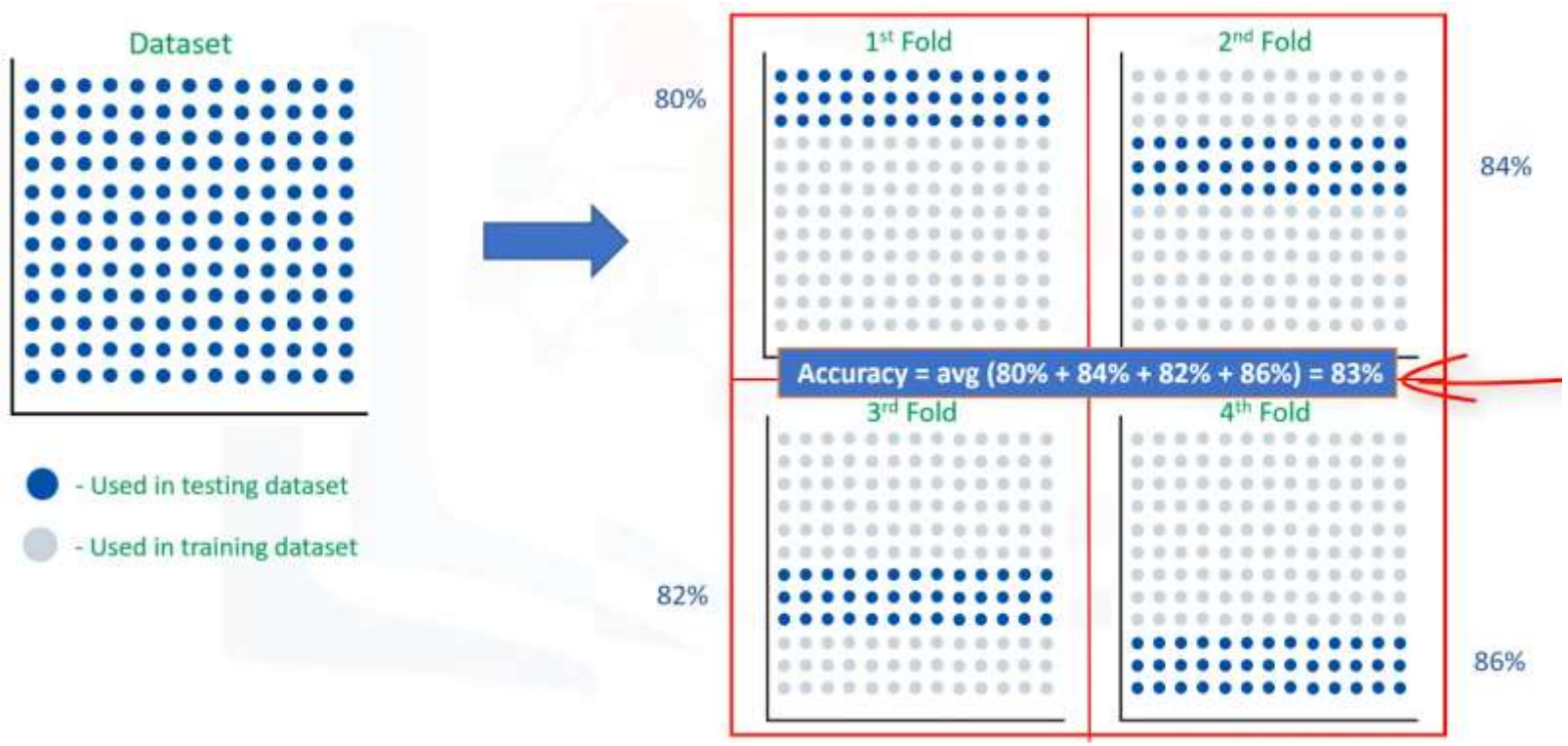
L'ensemble de données est représenté par les points de l'image en haut à gauche. Si nous avons K est égal à quatre fois, alors nous divisons cet ensemble de données comme indiqué ici. Dans le premier volet, par exemple, nous utilisons les premiers 25 % de l'ensemble de données pour les tests et le reste pour la formation. Le modèle est construit à l'aide de l'ensemble d'apprentissage et est évalué à l'aide de l'ensemble de test. Ensuite, au tour suivant ou au deuxième volet, les 25 % restants de l'ensemble de données sont utilisés pour les tests et le reste pour l'entraînement du modèle. Là encore, la précision du modèle est calculée. On continue pour tous les plis. Enfin, le résultat des quatre évaluations est moyenné. C'est-à-dire que la précision de chaque pli est ensuite moyennée, en gardant à l'esprit que chaque pli est distinct, où aucune donnée d'apprentissage dans un pli n'est utilisée dans un autre.

La validation croisée K-fold dans sa forme la plus simple effectue plusieurs fractionnements de train/test, en utilisant le même ensemble de données où chaque division est différente. Ensuite, le résultat est moyen pour produire une précision hors échantillon plus cohérente.

Example cross-validation



Example of cross-validation

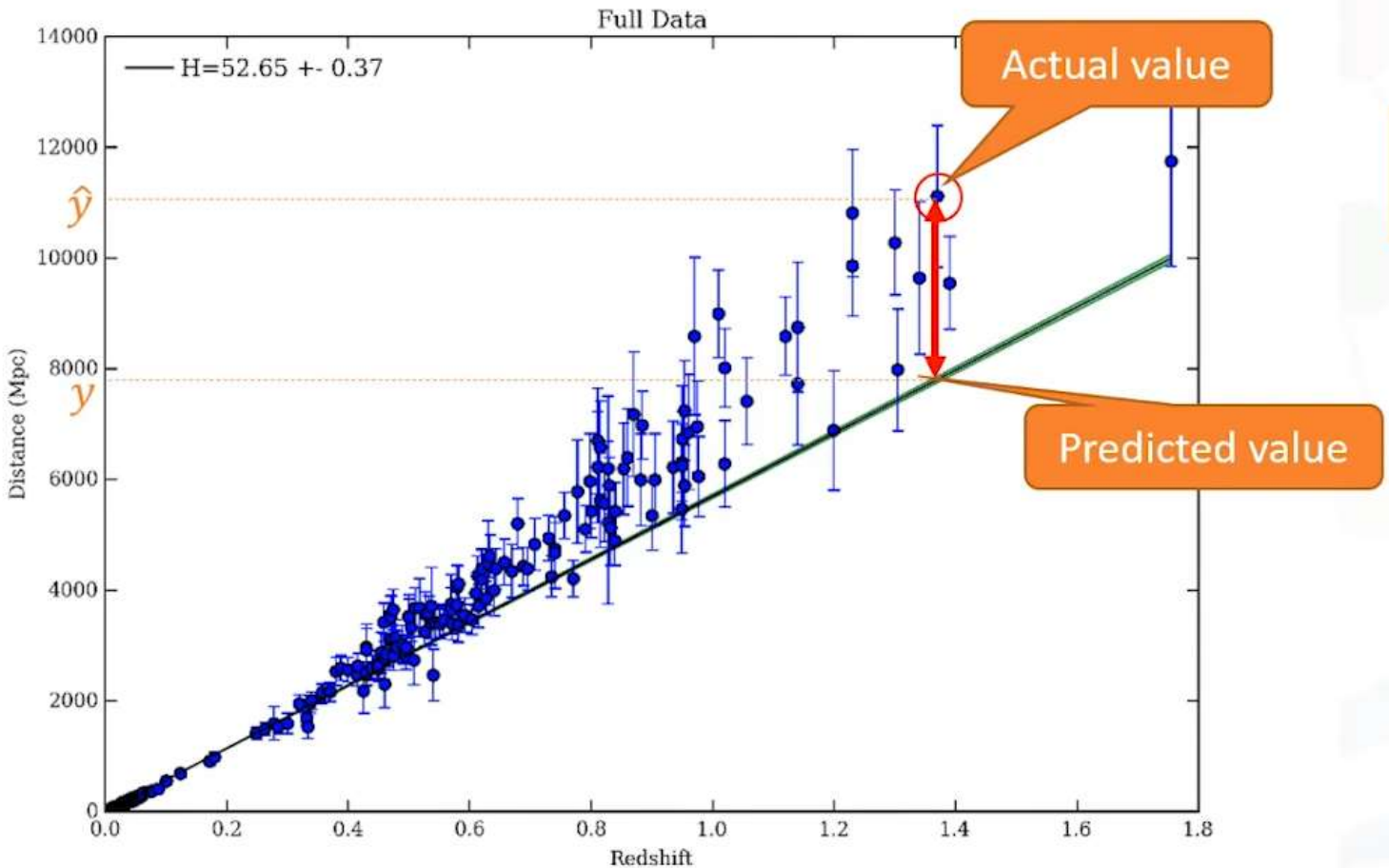


Evaluation Metrics in Regression Models

Les métriques d'évaluation

Les métriques d'évaluation sont utilisées pour expliquer les performances d'un modèle. pour calculer la précision de notre modèle de régression. Les mesures d'évaluation jouent un rôle clé dans le développement d'un modèle, car elles donnent un aperçu des domaines qui nécessitent des améliorations. Nous passerons en revue un certain nombre de métriques d'évaluation de modèle, notamment **l'erreur absolue moyenne**, **l'erreur quadratique moyenne** et **l'erreur quadratique moyenne(**root mean squared error**).** Mais avant de commencer à les définir, nous devons définir ce qu'est réellement une erreur. Dans le contexte de la régression, l'erreur du modèle est la différence entre les points de données et la ligne de tendance générée par l'algorithme.

Evaluation metrics are used to explain the performance of a model. to calculate the accuracy of our regression model. Evaluation metrics, provide a key role in the development of a model, as it provides insight to areas that require improvement. We'll be reviewing a number of model evaluation metrics including mean absolute error, mean squared error, and root mean squared error. But before we get into defining these, we need to define what an error actually is. In the context of regression, the error of the model is the difference between the data points and the trend line generated by the algorithm.



Error: measure of how far the da
is from the fitted regression line

	ENGINESIZE	CYLINDERS	FUELCONSUMPTION_COMB	CO2EMISSIONS
0	2.0	4	8.5	196
1	2.4	4	9.6	221
2	1.5	4	5.9	136
3	3.5	6	11.1	255
4	3.5	6	10.6	244
5	3.5	6	10.0	230
6	3.5	6	10.1	232
7	3.7	6	11.1	255
8	3.7	6	11.6	267
9	2.4	4	9.2	212

Test

y

Actual values

$$Error = \frac{(232 - 234) + (255 - 256) + \ ...}{4}$$

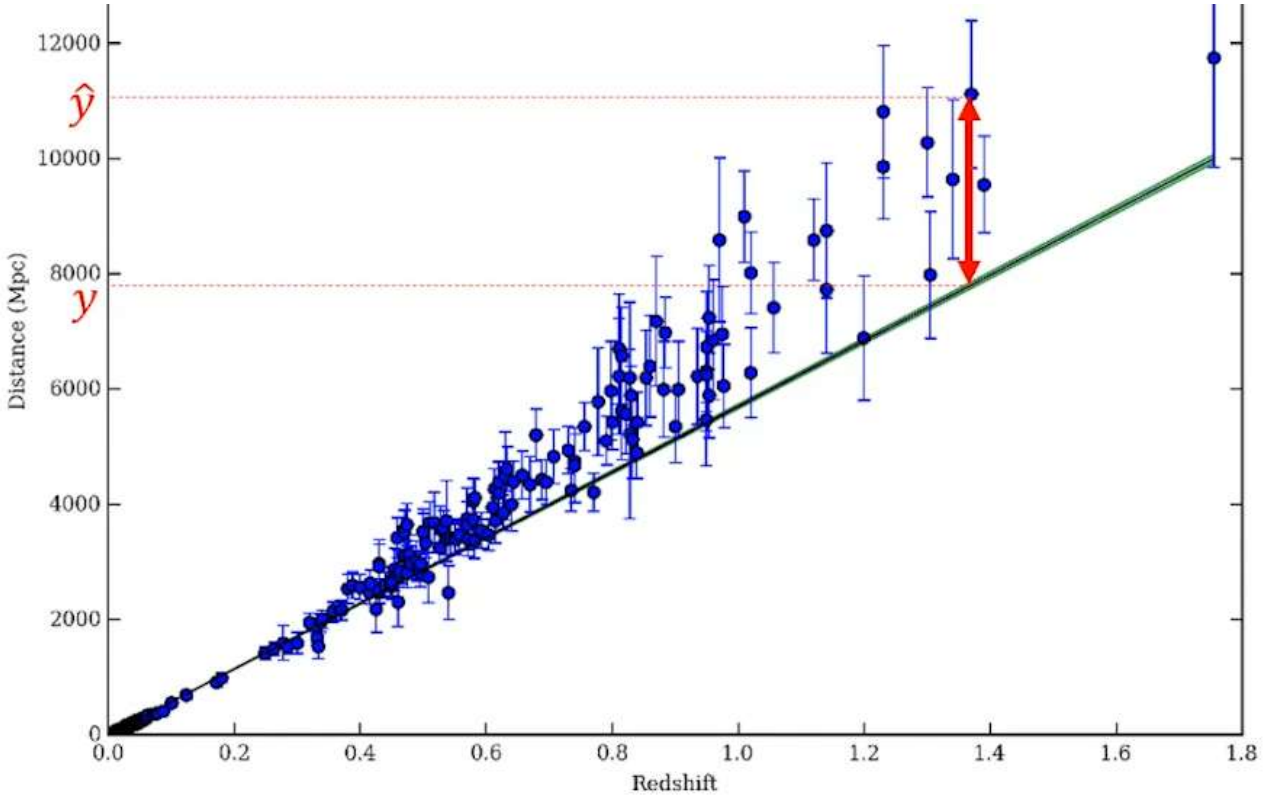
$$Error = \frac{1}{n}\sum_{j=1}^{n}|y_j - \hat{y}_j|$$

$\hat{y}$

	Prediction
6	234
7	256
8	267
9	210

Predicted values

Errors



$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$$

$$\mathrm{R}MSE = \sqrt{\frac{1}{n}\sum_{j=1}^{n}(y_j - \hat{y}_j)^2}$$

$$RAE = \frac{\sum_{j=1}^{n}|y_j - \hat{y}_j|}{\sum_{j=1}^{n}|y_j - \bar{y}|}$$

$$RSE = \frac{\sum_{j=1}^{n}(y_j - \hat{y}_j)^2}{\sum_{j=1}^{n}(y_j - \bar{y})^2}$$

$$R^2 = 1 - RSE$$

Since there are multiple data points,an error can be determined in multiple ways.Mean absolute error is the mean of the absolute value of the errors. This is the easiest of the metrics to understand, since it's just the average error.Mean squared error is the mean of the squared error.

It's more popular than mean absolute error because the focus is geared more towards large errors. This is due to the squared term exponentially increasing larger errors in comparison to smaller ones. Root mean squared error is the square root of the mean squared error. This is one of the most popular of the evaluation metrics because root mean squared error is interpretable in the same units as the response vector or y units, making it easy to relate its information.

Relative absolute error, also known as residual sum of square, where y bar is a mean value of y, takes the total absolute error and normalizes it by dividing by the total absolute error of the simple predictor. Relative squared error is very similar to relative absolute error but is widely adopted by the data science community, as it is used for calculating R squared. R squared is not an error perse but is a popular metric for the accuracy of your model. It represents how close the data values are to the fitted regression line. The higher the R-squared, the better the model fits your data.

Questions?