Part 1 Sampling

Q 1.1

Every morning, the student wakes up and randomly samples from this distribution an activity to do for the rest of the day. Provided that you can only sample from uniform distribution over (0,1), write a pseudocode to sample from the given multinomial distribution.

Connaissant les différentes probabilités des différentes activités que l'étudiant peut choisir, on peut établir des intervalles entre 0 et 1 avec ceux-ci. En utilisant une fonction de distribution uniforme en 0 et 1, on peut échantillonner celle-ci et trouver dans qu'elle intervalle se situe l'échantillon. La réponse nous donnera alors ce que l'étudiant fera dans sa journée. Cet algorithme s'exécute en O(n) ce qui est très bon, mais cela reste la méthode brute pour résoudre le problème. En effet, il existe une autre façon qui est utile lorsque l'on possède un très large jeu de donnée. Cette méthode consiste à construire une table d'échantillonnage en O(n log n) et après échantillonné celle-ci par la suite ce qui se fait en O(1).

```
"""""""""
p_values = [0.1, 0.6, 0.3]

create cdf list from p_value

sample form a uniform dist

for each element of cdf
    if u < element of cdf
        return index of element
"""""""""
```

Q 1.2

Implement your sampling algorithm and use it to sample the student's routine for 100 days. Report the fraction of days spent in each activity. Now use it to sample for 1000 days. Report the fraction of days spent in each activity. Compare these fractions to the underlying multinomial distribution.

Cette méthode nous permet de simuler une distribution multinomiale à l'aide d'une distribution uniforme. On voit que les valeurs obtenues en exécutant cette simulation plusieurs fois nous permet d'obtenir des valeurs proches des valeurs de départ. De plus, on voit que plus le nombre d'essai est élevé, plus les valeurs se rapproche de la distribution.

```
For 100 days
 - Movies: 0.17%
 - INF8245E: 0.44%
 - Playing: 0.11%
 - Studying: 0.28%

For 1000 days
 - Movies: 0.187%
 - INF8245E: 0.411%
 - Playing: 0.084%
 - Studying: 0.318%
```

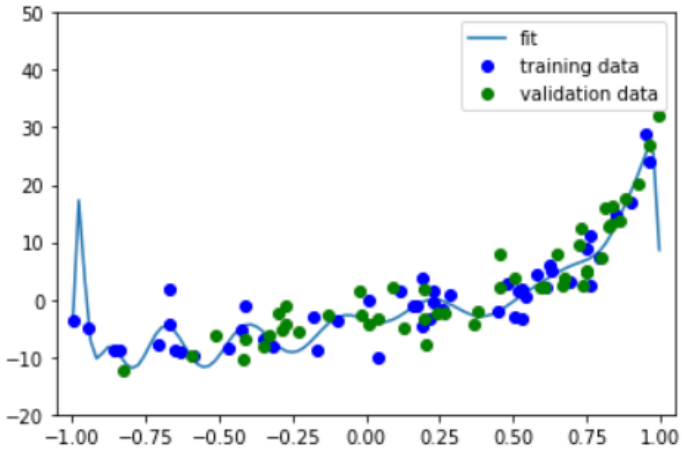
Part 2 : Model Sélection

Q 2.1 : Fit a 20-degree polynomial to the data.

(a) Report the training and validation RMSE (Root Mean-Square Error). Do not use any regularization.

```
RMSE - Training set:   2.6725914866203953
RMSE - Validation set: 4.934534526714249
```

b) Visualize the fit.



(c) Is the model overfitting or underfitting? Why?

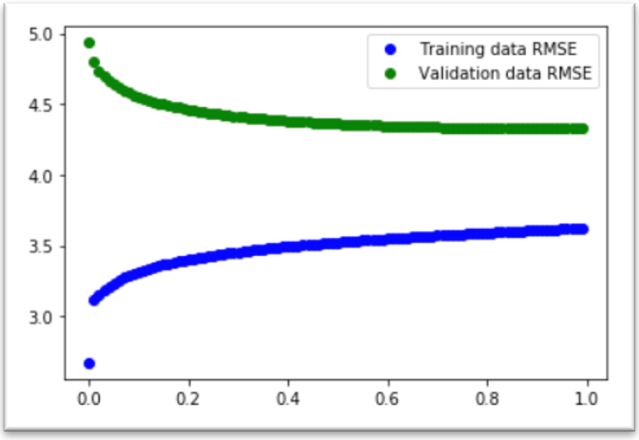
Le model over fit :

- RMSE : Le rmse de validation est deux fois plus élevé de celui de validation. Cela nous indique que le model over fit

- Graphique : on s'aperçoit rapidement que la courbe générée essai le plus possible de passer par tous les point du jeu d'entrainement. On voit cet effet de façon très amplifié sur les deux plus petites données du jeu d'entrainement, on voit la courbe faire un gros pique afin de pouvoir passer sur ces deux points. De plus, on remarque que la courbe zigzag dans le milieu du jeu de données, lorsque celle-ci devrait plutôt passer de façon droite et, finalement, on voit que la courbe descend vers le bas à la fin des données alors que celle-ci devrait continuer à augmenter pour pouvoir être une bonne extrapolation et pouvoir passer par les données de validation.

Q 2.2 : Now add L2 regularization to your model. Vary the value of λ from 0 to 1, with a 0.01 step size.

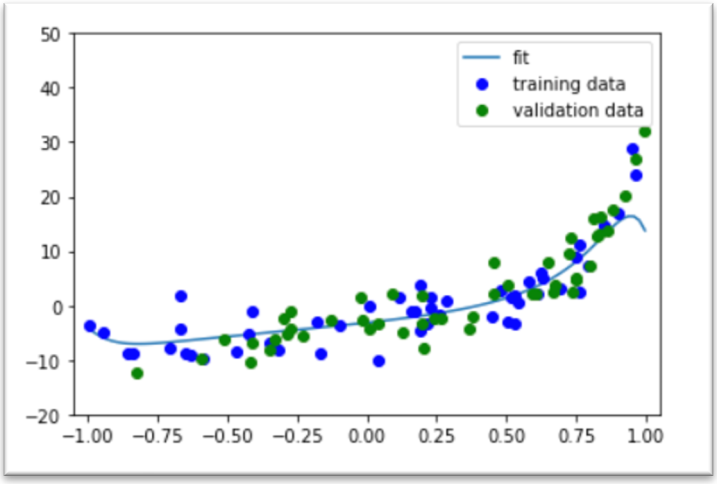
(a) For different values of λ, plot the training RMSE and the validation RMSE.



(b) Find the best value of λ and report the test performance for the corresponding model.

```
The best λ found based on the rmse of the validation test
 λ: 0.99
 RMSE-V: 4.327222332330323
```

(c) Visualize the fit for the chosen model.



(d) Is the model overfitting or underfitting ? Why?

Le model underfit un petit peut. En effet, il évalue moins bien les plus grandes données, car il la courbe cherche à descendre au lieu de monter. Le meilleur lambda serait alors un lambda intermédiaire. Il serait aussi pertinent de tester des model polynomial à degré plus petit pour se jeu de données

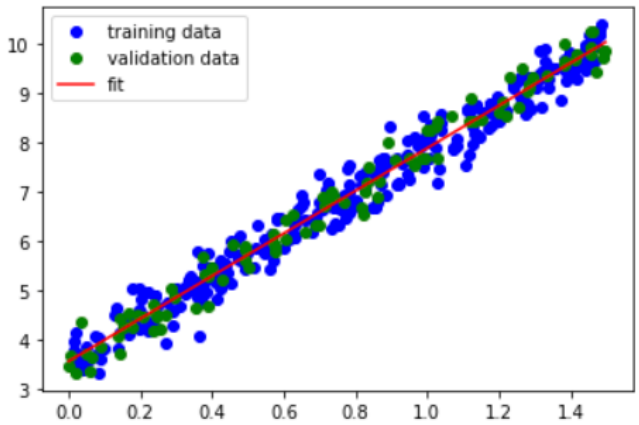
Q 2.3 What do you think is the degree of the source polynomial? Can you infer that from the visualization produced in the previous question?

D'après moi, le dégrée de la courbe polynomiale source est de 4 ou 6 selon le graphique. En effet, la courbe est croissant des deux côtés et la distribution fonctionnerait pour ces deux degrées.

Part 3 : Gradient Descent for Regression

Q 3.1 : Fit a linear regression model to this dataset by using stochastic gradient descent

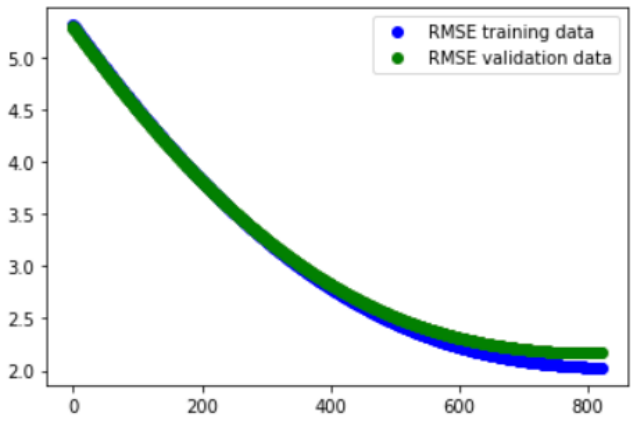
En utilisant la descente de gradient stochastique, et en considérant la convergence comment étant le moment ou le rmse de jeu de validation comme à monter, on obtient le fit suivant



(a) Using a step size of 10−4, plot the training and validation RMSE against the number of epochs, until convergence.

Lorsque l'on regarde les différent rmse des deux modèles, on s'aperçoit que les rmse commence par descendre à un rythme quasi similaire et que le rmse du jeu d'entrainement descend d'avantage que celui du jeu de validation vers la fin. Cette différence n'est toutefois pas grande car les données se ressemble grandement et l'on essaie de faire fit une droite au travers de ceux-ci alors le model ne peut pas vraiment overfit les données d'entrainement.

```
Final RMSE validation: 0.2721417485493501
Final RMSE test: 0.26311734625718175
```



Q 3.2: Try different step sizes and choose the best step size by using the validation data.

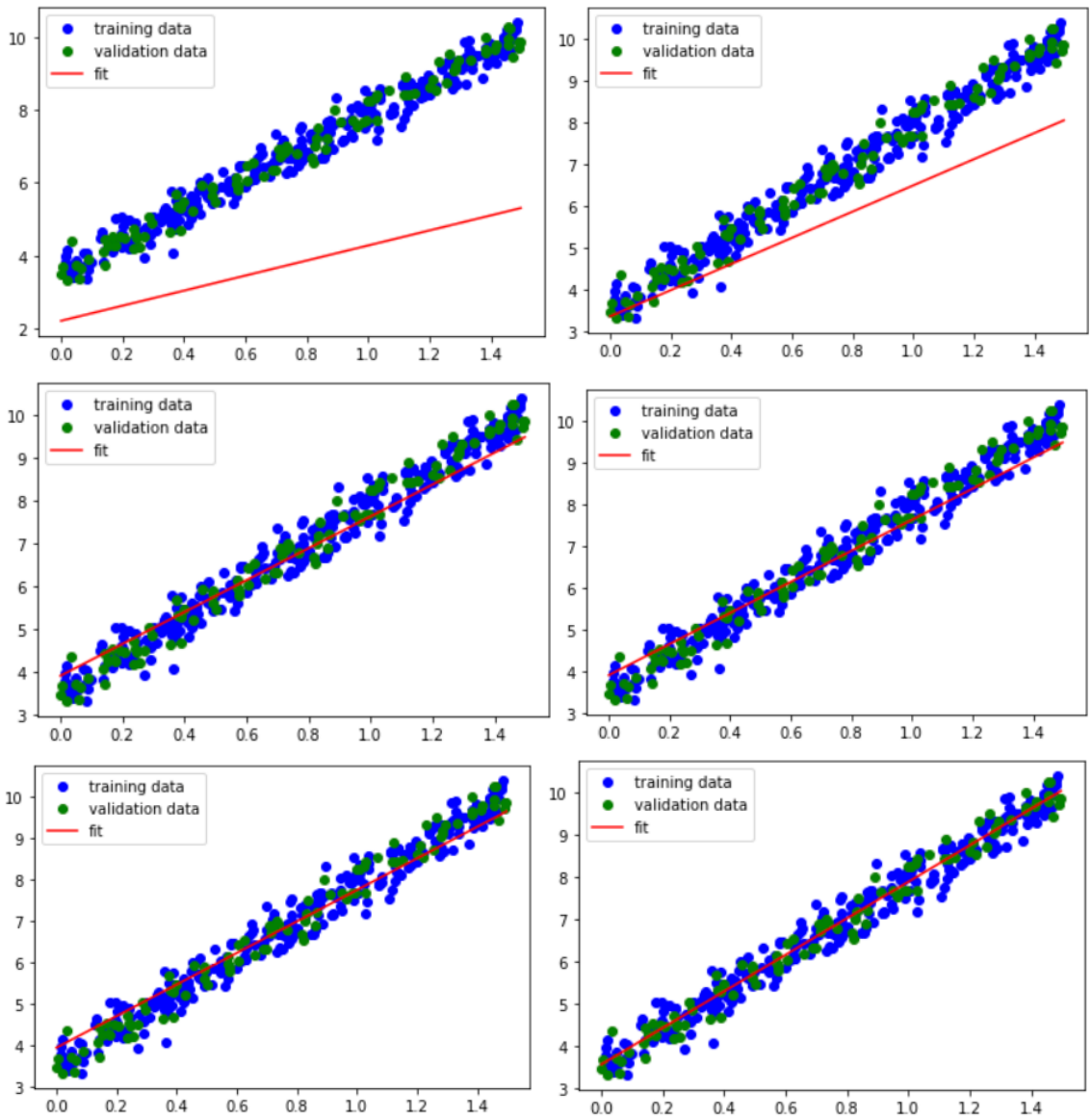
(a) Report in a table the validation performance with different step-sizes.

Learning_rate,	rmse_validation
0.001	0.272004707467
0.0001	0.272141748549
0.00001	0.272157034142

(b) Report the test RMSE of the chosen model.

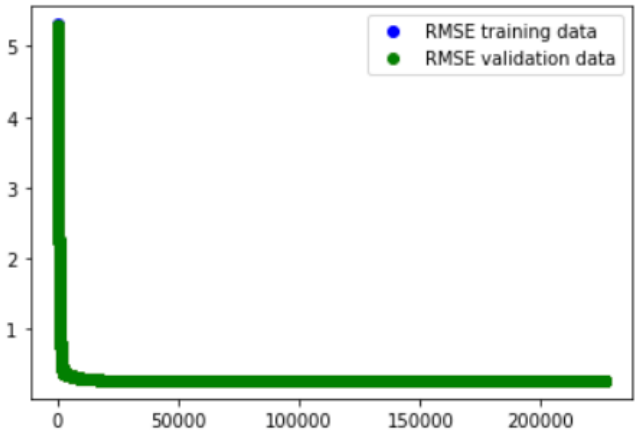
Le meilleur model est celui avec un learning_rate de 0.01. Celui-ci à permis d'obtenir le plus petit rmse sur le jeu de données de validatoin.

Q 3.3 Report 5 different visualizations chosen at random to illustrate how the regression fit evolves during the training process.



Q 3.4 Repeat part 1 using full-batch gradient descent.

```
rmse validation (final): 0.27215874364161285
```



Pour le Full batch gradient descent, on remarque que pour obtenir un rmse similaire à celui obtenue à l'aide de la méthode stochastique, il nous a fallu beaucoup plus d'epoche et de temps de calcul.

Q3.5 Comment on the difference between full-batch gradient descent and stochastic gradient descent based on your experiments.

Pour faire fit un modèle de régression linéaire sur les données, la méthode du gradient stochastique est beaucoup plus rapide. Mais pour cela, il est important d'avoir un learning_rate approprié. En effet, lorsque celui-ci est d'un ordre de grandeur trop petit ou trop grand, le temps es

Part 4 : Real lie dataset

Q 4.1 Make the data set usable

(a) Use the sample mean of each column to fill in the missing attributes. Is this is a good choice? Explain why or why not.

> Bien qu'utilisé la moyenne est une façon simple de compléter les valeurs manquantes et que cela peut bien fonctionner pour de petits jeux de données, ce n'est pas la meilleure des méthodes. Tout d'abord, cette méthode ne fonctionne pas pour les valeurs non numériques comme la colonne communityname dans notre jeu de données. De plus elle ne fait pas de sens pour les colonnes qui sont des catégories encoder en nombre comme plusieurs de nos colonnes. Finalement, la moyenne ne tient pas compte de la corrélation entre les différentes colonnes.

(b) What else might you use to fill in the missing attributes?

> Pour combler les données manquantes, on pourrait utiliser la valeur la plus fréquent pour chaque colonne, ou simplement mettre ces données à 0. Cependant, ces façons de faire ne tienne toujours pas compte de la corrélation entre les colonnes. Pour tenir compte de la corrélation entre les colonnes, une technique intéressante est de prédire les valeurs des données manquant en utilisant k nearest neighbours se basant sur les features similaires des différents points.

(c) If you have a better method, describe it, and use it for filling in the missing data. Explain why your method is better.

> Technique utilisée : la valeur la plus fréquent pour chaque colonne.

(d) Turn in the completed data set.

Q 4.2 Use the first 20% of the dataset for testing and use the remaining 80% for training in the order given in the dataset file.

(a) Report the 5-fold cross-validation average RMSE.

(b) Report the test RMSE.

Q 4.3 We now use Ridge-regression on the above data.

(a) In order to choose the best λ, plot the average RMSE using 5-fold cross validation, for various values of λ [x-axis: λ, y-axis: Average RMSE]. Explain how you chose the range of λ to explore.

> En augmentant lambda, on diminue la variance, mais augmente le biais du model. Il faut alors choisir de petite valeur de lambda pour ne pas rendre le model trop simple.

(b) Which value of λ gives the best fit?

(c) Report the test RMSE using the value of λ you chose.

(d) Is it possible to use the information obtained during this experiment for feature selection? If so, explain how?

In

(e) Report the test RMSE of the best fit you achieve with a reduced set of features?

(f) How different is the performance of the model with reduced features compared to the model using all the features? Comment about the difference.