

# Compreensão de padrões de aluguer de bicicletas na cidade de Washington D.C.

Diogo Borges — 49906  
Vasco Leitão — 49455

Complementos de Aprendizagem Automática  
FACULDADE DE CIÊNCIAS DA UNIVERSIDADE DE LISBOA

May 30, 2020

# 1 Introdução

Nos tempos que correm, procuram-se novas formas de desenvolver meios de transporte públicos que não interfiram de forma significativa sobre a topologia e o trânsito das cidades modernas. Os sistemas de *bike-sharing* oferecem uma solução moderna, prática e com pouco impacto sobre a pegada ecológica de uma cidade.

Existem dois tipos de sistemas existentes (1): sistemas desenvolvidos por organizações ou grupos comunitários, e sistemas criados pelos municípios, em parcerias com empresas que também procuram maximizar o lucro através da implementação do sistema.

Um problema relativamente complexo que estes sistemas apresentam é entender quais os pontos de interesse para alocar os *dockers*, onde são colocadas as bicicletas, e perceber se estas localizações têm procura elevada. A procura elevada leva a que se tenha de disponibilizar mais bicicletas para os utilizadores, mas inicialmente compreender os padrões que ocorrem temporalmente e termos de localização, particularmente em relação à quantidade de bicicletas alugadas.

## 2 Abordagem

O objetivo deste trabalho é promover um modelo de previsão de número de bicicletas alugadas para um sistema de *bike-sharing*.

Para construção e treino do modelo optámos por usar um *dataset* que agrupa todas as viagens de bicicleta registadas durante janeiro de 2020 (dia 1 até dia 31, inclusive), na cidade de Washington D.C., situada nos Estados Unidos da América, sendo estes dados fornecidos publicamente através da empresa *Capital Bikeshare*.

O *dataset* divide-se essencialmente por quatro tipos de dados:

1. **Informação temporal:** informação relativa a data e hora de início e de fim de viagem, além da duração da viagem;
2. **Informação geográfica:** informação das estações de início e fim de viagem em relação à latitude, longitude e capacidade máxima de bicicletas alocadas;
3. **Informação organizacional:** informação relativamente ao número da bicicleta e o tipo de cliente que efetuou a viagem;
4. **Informação meteorológica:** dados relativos ao dia de início de viagem em termos de: temperatura (mínima, média, máxima); velocidade do vento; precipitação.

### 2.1 Pré-processamento dos dados

O pré-processamento dos dados divide-se por vários pontos: recolha dos registo, obtenção das localizações de cada estação, caracterização do clima para cada um dos dias do período de tempo onde se insere o problema, e definição do tipo de dia.

### 2.2 Tratamento de dados geográficos

Algoritmos de *clustering* procuram fazer a identificação em termos de classes para um conjunto espacial. De modo a conseguir usar os dados geográficos de forma mais eficaz no modelo, definimos como *sub-task* o uso de algoritmos de *clustering* de modo a agrupar as localizações das estações por zonas. Propomos três aplicações diferentes de modelos de *clustering*, descritas por *Xu et al.* (3) e *Dueck et al.* (4): *distance-based*, *density-based*, e *affinity-based*, de modo a entender quais são aqueles que oferecem melhores resultados. Como objetivo desta *sub-task*, queremos entender se as *labels* definidas através do *clustering* oferecem mais informação para os modelos de regressão, através de análise exploratória. *Vogel et al.* (5) apresentam propostas interessantes em termos da análise destes modelos.

## 2.3 Aplicação de modelos de regressão

Os variados modelos de regressão têm como objetivo fazer uma previsão de forma precisa do número de alugueres por hora, ao longo do mês, para cada uma das zonas definidas através dos modelos de *clustering*. Dado um conjunto-base e as previsões com esses regressores, será escolhido o ”melhor” modelo através da análise de métricas. A partir desse modelo, serão feitas previsões de múltiplas formas, para entender do que depende a sua capacidade de previsão (recorrendo a visualizações), e quais são os pontos que tem maior impacto nos resultados.

# 3 Implementação

## 3.1 Pré-processamento dos dados

O pré-processamento é o primeiro passo do processo e talvez o mais essencial para garantir a legibilidade dos resultados. Para este caso em específico tivemos de recorrer a variadas fontes e incorporá-las para construir o *dataset* descrito na secção ‘Abordagem’. Como base temos um *dataset* com um conjunto de registos relativamente a viagens efetuadas em janeiro de 2020 (6), com informações relativamente às estações que definem o ponto inicial e final da viagem e os horários das viagens. Para complementar este conjunto, obtivemos as localizações das estações - também disponíveis de forma pública - e a informação relativa à meteorologia (7) para cada dia do período temporal do problema.

A definição da *feature* para distinguir se um dado dia é dia útil ou não foi considerada relevante para entender se influenciava o número de alugueres. Para tal iremos criar uma coluna que representa se é um dia de trabalho ou não (feriado ou fim de semana). Tendo em conta os feriados em Janeiro de 2020 em Washington iremos considerar como dias não úteis o dia 1-1-2020 (New Year’s Day) e o dia 20-1-2020 (Martin Luther King Jr. Day)(8).

## 3.2 Tratamento dos dados geográficos

A aplicação dos modelos de *clustering* baseia somente na localização de cada estação e das *features* de latitude e longitude.

Para esta *sub-task* propomos a análise dos *clusters* formados por três tipos de modelos, visto que oferecem perspetivas diferentes relativamente ao problema:

- *Squared-error-based* (ou *distance-based*): **K-Means** - o objetivo é entender se o recurso à distância euclidiana permite identificar *clusters* e centróides relativos às diferentes regiões da cidade;
- *Density-based*: **DBSCAN** (9) - propósito de perceber se a distribuição das estações pela cidade define diferentes *clusters* bem separados e representativos;
- *Affinity-based*: **Affinity Propagation** - análise de *clusters* sem qualquer necessidade de recorrer a um valor de K pré-definido, procurando uma forma natural de processar a similaridade entre estações.

A análise destes modelos vai depender das métricas de **Silhouette Score**, **Calinski-Harabasz Index** e **Davies-Bouldin Index**, complementada pela análise gráfica da separação dos *clusters*. Após a escolha do ”melhor” modelo, prosseguiremos à análise dos *clusters* formados e dos padrões que observam entre os mesmos.

## 3.3 Aplicação dos modelos de regressão

Para o ponto principal deste trabalho iremos procurar prever a procura de bicicletas num dado dia, numa dada hora e numa dada zona que será dada pela análise de *clustering*. Existe uma preparação necessária dos dados porque é necessário um agrupamento dos mesmos consoante o referido anteriormente e depois contar o número de alugueres. É também necessário reter apenas as variáveis de interesse para a análise.

Recorrendo a vários modelos avaliar o que se adequa mais (usando métricas) e adaptando parâmetros para ser usado nas previsões futuras. Como dados de treino irão se usar os primeiros 21 dias do mês e como dados de teste os últimos 10 dias.

Vamos recorrer a diferentes tipos de modelos, analisando métricas como o RMSE e R<sup>2</sup>:

- *Ensemble estimators*: **Random Forest Regressor**;
- *Boosting estimators*: **AdaBoost Regressor**;
- *Bagging estimators*: **Bagging Regressor**;
- **Support Vector Regressors e K-Nearest Neighbors**.

## 4 Resultados

### 4.1 Pré-processamento dos dados

Não vamos dar particular atenção a esta subsecção mas queremos explicar alguns pontos que são relevantes para os resultados do modelo. Alguns registos não tinham disponíveis dados relativamente a localizações, visto que não foi possível fazer o mapeamento entre determinadas estações e a sua localização, por isso estes registos foram removidos, uma vez que são reduzidos comparativamente ao número total de registos no conjunto. De referir que os dados já tinham algum pre-processamento inicial em que foram removidas viagens realizadas por staff, e viagens que duraram menos de 60 segundos.

### 4.2 Tratamento dos dados geográficos através de *clustering*

Ao iniciar-se o processo de *clustering* primeiro viu-se a disposição das docas de acordo com a Latitude e Longitude. Como é possível verificar a partir da **Figura 1** temos representado um centro com alta densidade de estações e à medida que se vai afastando deste centro, há uma clara redução desta densidade, salvo alguns centros que concentram determinados grupos de estações.

#### 4.2.1 KMeans

O primeiro modelo a ser testado foi o KMeans, fazendo variar o  $k$ , referente ao número de *clusters*, obtendo os resultados descritos nas **Figuras 2, 3**.

Embora se obtenham valores satisfatórios em cada métrica, em termos práticos as fronteiras não são bem definidas, existindo pouca distância *intra-cluster*, sendo que o modelo iria ser bastante sensível e interpretar valores diferentes para estações que se encontram próximas.

#### 4.2.2 DBSCAN

No DBSCAN, e dadas as variações de latitude e longitude entre as diferentes estações, variou-se  $\epsilon \in [0.025, 0.027, 0.029]$  e  $N_{min} \in [2, 4]$  de forma a observar-se a variação dos resultados, descritos nas **Figuras 4, 5**. Os clusters gerados pelo DBSCAN parecem ser muito mais naturais que os produzidos pelo KMeans. Este conceito de naturalidade está associado à distância entre os clusters obtidos, que é muito maior que no K-Means e leva a fronteiras muito melhor definidas. Dependendo do número de clusters produzidos conseguem-se distinguir facilmente diferentes zonas existindo um cluster de grande dimensão colorido a vermelho e vários outros de menor dimensão mas muito bem definidos.

Observando diretamente os resultados do DBSCAN, também é possível identificar determinadas combinações dos hiperparâmetros (nas quais  $N_{min}$ ) que não definem ruído, sendo que essas combinações oferecem um agrupamento de localizações mais próximo da realidade, visto que os valores de *Davies-Bouldin Index* também se aproximam de 0. Sendo assim, queremos escolher uma combinação que não só maximize o *Silhouette Score* e o *Calinski-Harabasz Index* mas que particularmente tenha valores bastante reduzidos de *Davies-Bouldin Index*. Logo, a melhor caracterização dos *clusters* obtém-se com  $(\epsilon, N_{min}) = (0.027, 2)$ .

#### 4.2.3 Affinity Propagation

Em termos de parâmetros, apenas procuramos usar o *standard* definido pela biblioteca *Scikit-Learn*. Os resultados não são muito positivos, visto que o número de *clusters* criados, como é possível ver na **Figura 6**, é bastante elevado, onde cada *cluster* corresponde a um conjunto bastante reduzido de estações e que não aparentam ter qualquer distinção. Visto que provavelmente um número reduzido de estações por cada

*cluster* corresponderá a um número reduzido de alugueres dentro do mesmo, podemos logo descartar este modelo.

### 4.3 Implementação das *labels* de clustering e observação dos dados

Através da observação dos resultados da secção anterior, vamos fazer a análise dos dados a partir dos *clusters* criados através do **DBSCAN**, com parâmetros ( $\epsilon = 0.027, N_{min} = 2$ ), cuja visualização refere à **Figura 7**. Através da visualização é possível, desde logo, destacar determinadas zonas da cidade. Destaca-se a vermelho a zona central da cidade de Washington. Também é possível distinguir a amarelo uma região a norte a cidade de Rockville. A Oeste existem também 2 zonas distintas, a zona de Reston (mais à esquerda e azul claro) e a zona de Tyson's Corner (a verde). Existem outros pequenos grupos com apenas 3 estações, o que pode ser problemático em termos de reconhecimento de padrões de aluguer: Merrifield (a rosa) situada a sul de Tyson's Corner e Largo (a azul escuro) no lado mais Este da cidade.

Observou-se o número de viagens que foram feitas entre diferentes zonas (*clusters*) e apenas 40 ocorrem entre *clusters* diferentes. Um número muito reduzido de viagens entre *clusters* pode indicar uma boa divisão dos mesmos.

Decidimos criar um novo dataframe agrupando os dados por *cluster*, dia e hora e contando o número de alugueres para cada uma dessas combinações. Dado que temos **6 clusters, 31 dias e 24 horas** por dia, é esperado que se obtenham  $6 \times 31 \times 24 = 4464$  combinações possíveis.

Após fazer o agrupamento onde se considerou para análise as colunas referentes aos dias úteis, à temperatura média, à precipitação e à velocidade do vento constatou-se que das 4464 combinações possíveis, apenas 1470 apresentam valores que indicam aluguer de bicicletas, então vamos preencher todas as restantes com valores nulos, isto é, combinações nas quais não ocorrem quaisquer alugueres.

Referir que  $\frac{1470}{4464} \approx \frac{1}{3}$ . Dois terços das combinações não contêm qualquer registo, o que pode levar ao enviesamento dos resultados.

#### 4.3.1 Distribuição de alugueres pelos *clusters*

O senso comum indicará que quanto maior o número de *docks*, maior o número de alugueres. Observando a **Figura 8** é possível ver que a distribuição é bastante desnivelada, sendo que grande parte das *docks* se concentra no cluster 0 - o cluster central. Como esperado, a distribuição de alugueres ao longo do mês - **Figura 9** - é também ela bastante desnivelada, sendo até ao nível dos milhares. O cluster 0 é dominador deixando os restantes sem qualquer expressão.

Os padrões que separam os alugueres ao longo do dia são semelhantes, embora este padrão apresente maior peso para o cluster central. Mais uma vez é possível ver (pela **Figura 10**) através do eixo y que o número médio de bicicletas alugadas é muito baixo à exceção daquelas no cluster 0.

Observando o *dataset* como um todo, o agrupamento dos valores pelos *workday* e os *non-workday*, definido na **Figura 11**, revela um padrão interessante, tendo influência no número de alugueres, havendo no geral um número menor de alugueres nos dias não úteis - exceptuando no fim-de-semana dos dias 11 e 12 de janeiro. Esta variável vai ter uma influência interessante nos resultados, como se poderá ver mais à frente.

Existem outras variáveis que poderão ter influência no número de alugueres, nomeadamente as que se referem a dados meteorológicos - temperatura, precipitação e velocidade do vento. O gráfico da **Figura 12** procuram alinhar a variação dos alugueres e dos dados meteorológicos, embora não se reconheça correlação forte entre os mesmos. Como seria expectável há uma correlação positiva para a temperatura (quando maior a temperatura mais alugueres são efetuados) e uma correlação negativa para a precipitação e velocidade do vento, ou seja quando chove mais e/ou o vento é mais forte há tendência para as pessoas não recorrerem a este tipo de transporte. No entanto, esta correlação não tem um impacto muito forte nos alugueres.

## 4.4 Previsão do número de bicicletas agrupadas

Dadas as observações feitas na secção anterior, vamos trabalhar com o *dataset* como um todo, descartando a informação oferecida pelos *clusters*. A escolha prende-se principalmente com o facto de *clusters* não-centrais terem poucas estações e consequentemente, poucos registo ou até zero registo para determinadas horas, o que influenciaria de forma negativa a precisão dos modelos. Em vez de recorrer a implementações para *train-test split* existentes, o nosso objetivo foi fazer as previsões para os últimos 10 dias do mês.

- Conjunto-treino (*train*): observações dos primeiros 21 dias do mês;
- Conjunto-teste (*test*): observações dos restantes dias, descartando o número de alugueres.

Os tipos de previsão efetuados foram os seguintes:

- Uma hora - para entender a capacidade do modelo identificar a variação de alugueres ao longo do dia;
- Um dia - para entender a capacidade do modelo identificar a variação de alugueres ao longo dos vários dias;
- Dez dias - para entender a capacidade do modelo identificar variação a nível semanal;
- Tendo conhecimento dos dados meteorológicos futuros - para compreender se mais informação relativamente à meteorologia leva a maior precisão.

Fomos avaliar um conjunto de modelos consoante duas métricas: *Root Mean Squared Error* e  $R^2$ . O objetivo é apresentar um modelo que minimize RMSE e cujo  $R^2$  se aproxime de 1, sendo estabelecida dos mesmos nas **Figuras 13, 14**.

O modelo **RandomForestRegressor** foi aquele que obteve melhores resultados globalmente e como tal será este o modelo a ser usado. De notar que o **BaggingRegressor** mostrou-se também eficaz e por isso seria uma opção válida para a previsão dos alugueres.

Aplicou-se então o modelo aos diferentes conjuntos de dados de modo a fazer as previsões pré-estabelecidas e verificando a qualidade das mesmas de forma gráfica, já sabendo *a priori* os valores das métricas referidas, comparando as previsões com a variação real dos alugueres. Também se recorreu a implementações de *grid-search* de modo a determinar os parâmetros *optimal* para este modelo de regressão e conjunto de dados. De uma forma geral os resultados são positivos, sendo que o modelo tem facilidade em identificar, com alguma precisão, as tendências no número de alugueres. No entanto, comparando os diferentes modelos, é possível observar diferenças nos valores das métricas que devem ser referidas e que são suportadas pelos gráficos da **Figura 15**.

- *One-hour predictions*: resultados positivos, embora o modelo procure fazer o reconhecimento dos picos de *workdays* para os *non-workdays*, levando a resultados com maior erro.
- *One-day predictions*: de forma semelhante às *one-hour predictions*, o modelo não consegue captar os *non-workdays* e o número de registo não permite que ele se adapte a tempo. Estes dois modelos são relativamente fracos para compreender diferenças entre dias úteis e não-úteis.
- *Ten-days predictions*: Visto que o modelo passa a compreender as diferenças e as variações semanais, consegue antecipar a chegada do fim-de-semana, apresentando resultados muito mais precisos e um erro mais reduzido.
- Com previsões meteorológicas: Os resultados acabam por ser semelhantes às *one-hour predictions* visto que a influência que a meteorologia é relativamente alta, além de que a correlação entre essas *features* e *workdays/non-workdays* é residual.

## 5 Comentários Finais

Como referido no plano de trabalho, o objetivo era obter um modelo que não só conseguisse prever o número de alugueres de forma relativamente precisa, como esse modelo aproveitasse os dados geográficos que são disponibilizados publicamente. Recorrer apenas às *features* sobre a latitude e longitude para formar agrupamentos revela-se afinal um processo que é isolado da realidade que o *dataset* constitui.

O DBSCAN mostrou melhores resultados consistentes dada a capacidade de juntar estações com proximidade e criar zonas densas, acabando por gerar 6 clusters correspondentes às zonas de Washington D.C. e arredores. Esta separação era, à partida, tão prometedora, que apenas uma pequena fração das viagens é entre diferentes zonas. No entanto o que acaba por acontecer é que quase todas as viagens são realizadas na zona centro de Washington enquanto que outras partes têm baixa taxa de utilização, derivadas da fraca densidade de estações nessas zonas e da falta de ligações à zona central. Estes fenómenos acabavam por influenciar de forma negativa a previsão, sendo que o modelo não se conseguia adaptar a essa discrepância entre zonas. Logo, acabamos por descartar a categorização que os *clusters* ofereciam. Apesar disso, referir que apenas estamos a tratar dados relativos a um mês. Possivelmente, dados de um período temporal ligeiramente maior conseguem oferecer informação consistente à qual o modelo se consiga adaptar.

Observando as visualizações dos modelos regressivos, existem alguns padrões interessantes que podem ser analisados:

- A hora do dia é uma variável importante para explicar o número de alugueres, e espelha bem a realidade de uma cidade com características empresariais.
- Dias não úteis (ou *no working days*) são precisamente aquelas onde o fluxo é menor, derivado de menor movimentação relativamente ao trabalho. Explorando os fluxos nesses dias, existe maior movimentação durante a tarde.

Pegando neste último ponto também é possível verificar que o número de alugueres tende a ser maior nos dias úteis face a fins de semana e feriados.

Uma possível limitação que este modelo apresenta é a escolha do melhor modelo, visto que apenas depende da comparação dos modelos através dos parâmetros *standard*. Um processo extensivo de *hyperparameter tuning* com os vários modelos poderia provavelmente oferecer alternativas mais sólidas com resultados ainda mais concretos. Dos vários métodos testados o que foi melhor e se decidiu usar para as revisões foi o **Random Forest Regressor**, no qual os resultados acabaram por ser relativamente precisos, podendo concluir que o modelo poderá ser implementado futuramente em novos dados da empresa.

## 6 Bibliografia

1. Antoniades, P., and Chrysantho, A. "European best practices in bike sharing systems". T. aT.-Students Today, CitiZens Tomorrow (2009). 41. <http://scholar.google.com/scholar?hl=en&btnG=Search+q=intitle:European+Best>
2. Kaggle (2015), "Bike Sharing Demand (Competition)" . <https://www.kaggle.com/c/bike-sharing-demand/overview>
3. Xu, R., Wunsch, D. (2005). Survey of clustering algorithms. IEEE Transactions on Neural Networks, 16(3), 645–678. <https://doi.org/10.1109/TNN.2005.845141>
4. Frey, J., and Dueck, D. (2007), "Clustering by Passing Messages between Data Points", Science, vol. 315, 972-976, <http://utstat.toronto.edu/reid/sta414/frey-affinity.pdf>
5. Vogel, P., Greiser, T., Mattfeld, D.C. (2011). Understanding bike-sharing systems using data mining: exploring activity patterns. Procedia Soc. Behav. Sci. 20, 514–523
6. Capital Bikeshare, "System Data". <https://s3.amazonaws.com/capitalbikeshare-data/index.html>
7. POWER Data Access Viewer, NASA. <https://power.larc.nasa.gov/data-access-viewer/>
8. Office Holidays - Washington (2020). <https://www.officeholidays.com/countries/usa/washington/2020>
9. Ester, M., Kriegel, H., Sander, J., and Xu, X. (1996). "A Density-Based Algorithm for Discovering Clusters". <https://www.aaai.org/Papers/KDD/1996/KDD96-037.pdf>

## 7 Anexos

Todos estes anexos encontram-se disponíveis no material e implementação.

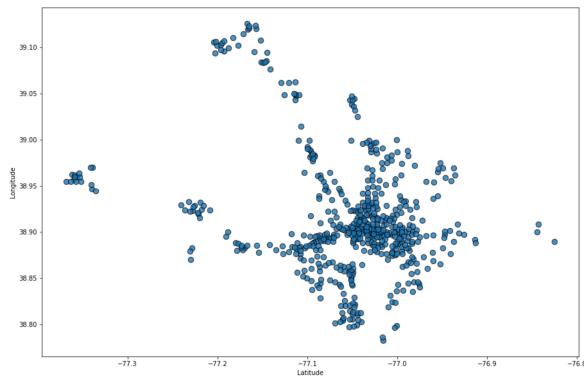


Figure 1: Distribuição de estações da cidade.

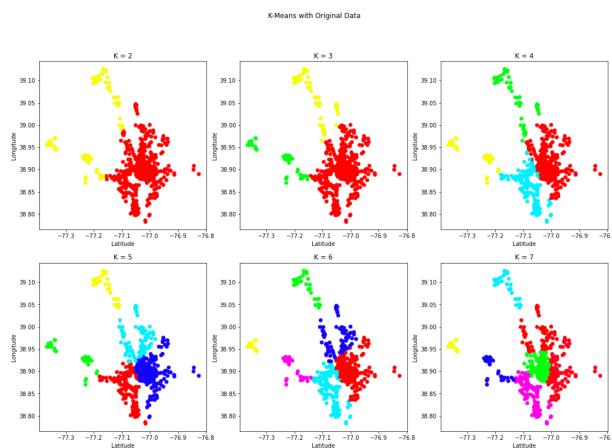
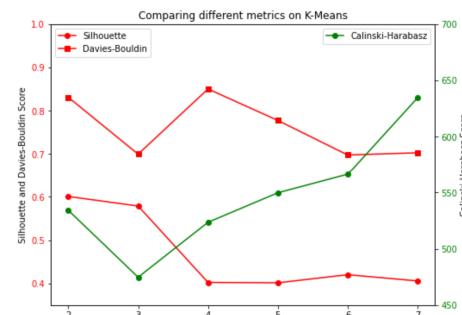


Figure 2: Distribuição de *clusters* para KMeans.

Value of K	Silhouette Score	Calinski-Harabasz Score	Davies-Bouldin Score
0	2	534.19310	0.63081
1	3	474.71285	0.69936
2	4	523.48646	0.84986
3	5	549.72301	0.77719
4	6	566.26035	0.69737
5	7	634.49061	0.70254

(a) Métricas de avaliação



(b) Evolução com variação de K

Figure 3: Comparação de métricas para KMeans.

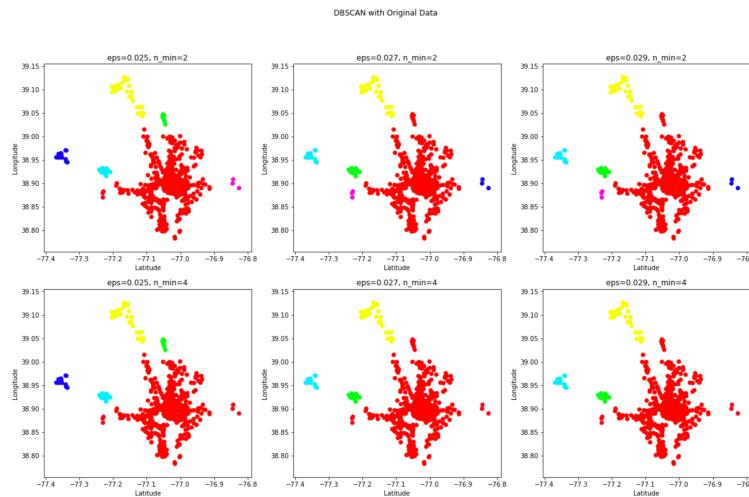
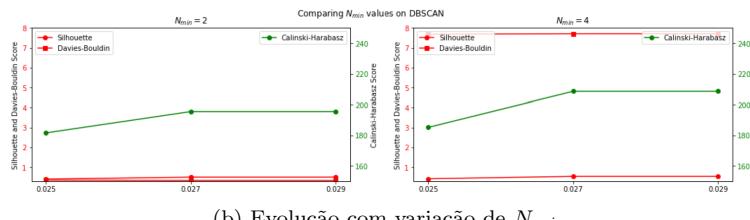


Figure 4: Distribuição de clusters para DBSCAN.

Epsilon	N_min	Silhouette Score	Calinski-Harabasz Score	Davies-Bouldin Score
0	0.025	2	0.40054	181.53104
1	0.027	2	0.49812	195.48503
2	0.029	2	0.49812	195.48503
3	0.025	4	0.41610	185.12019
4	0.027	4	0.53587	208.55462
5	0.029	4	0.53587	208.55462

(a) Métricas de avaliação



(b) Evolução com variação de N\_min

Figure 5: Comparação de métricas para DBSCAN.

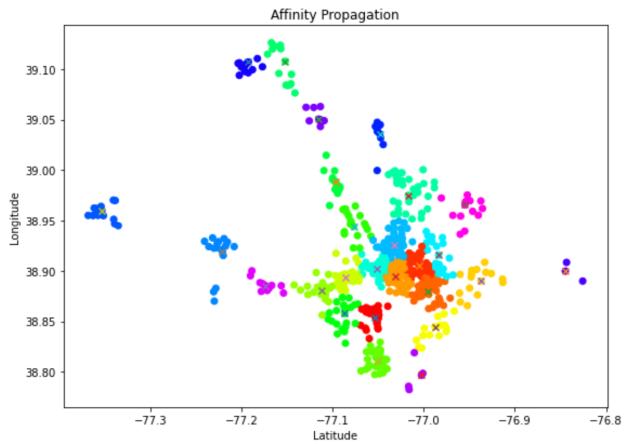


Figure 6: Distribuição de *clusters* para AP.

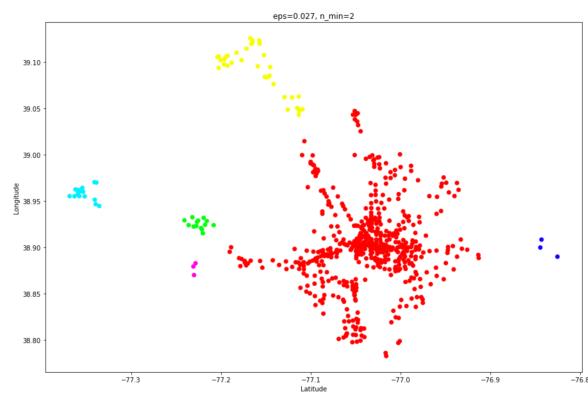


Figure 7: Distribuição de *clusters* para DBSCAN, com ( $\epsilon = 0.027, N_{min} = 2$ ).

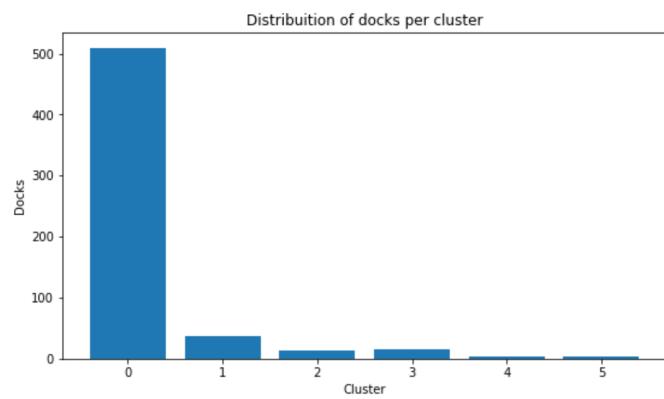


Figure 8: Distribuição de *docks* pelos *clusters*.

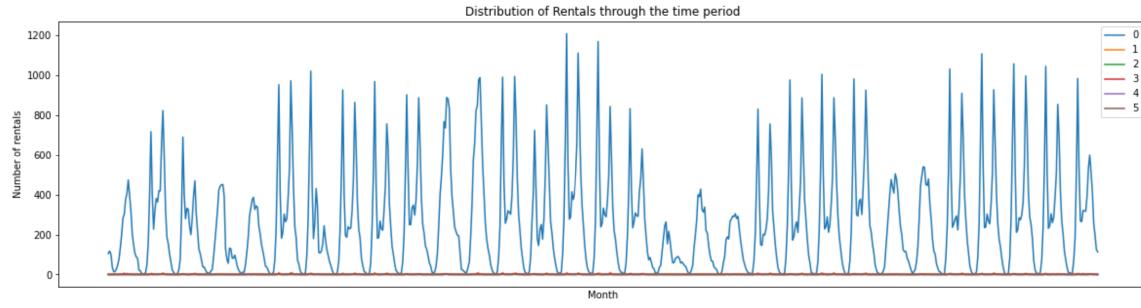


Figure 9: Distribuição de alugueres ao longo do mês.

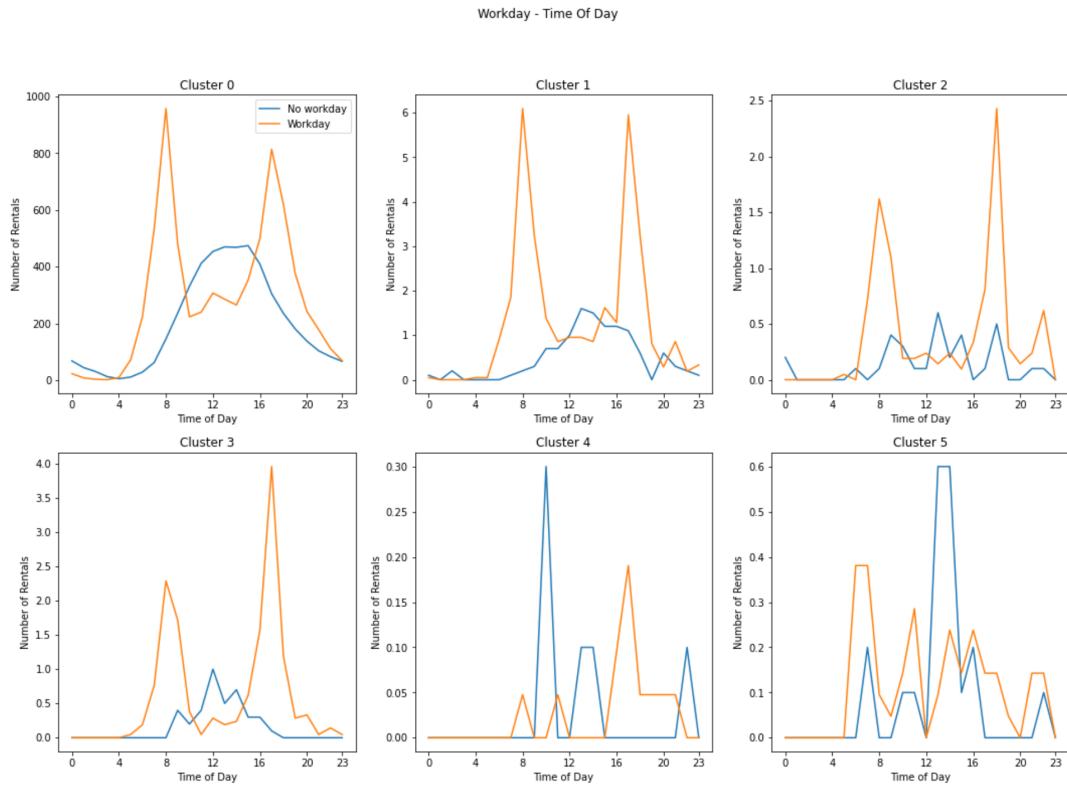


Figure 10: Distribuição de alugueres ao longo do dia para os vários clusters.

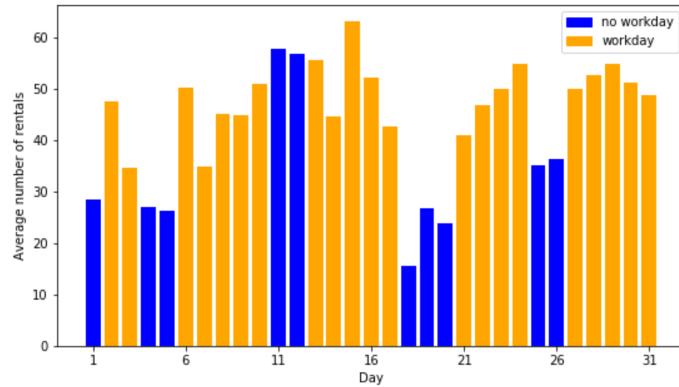


Figure 11: Distribuição de alugueres pelos vários dias do mês.

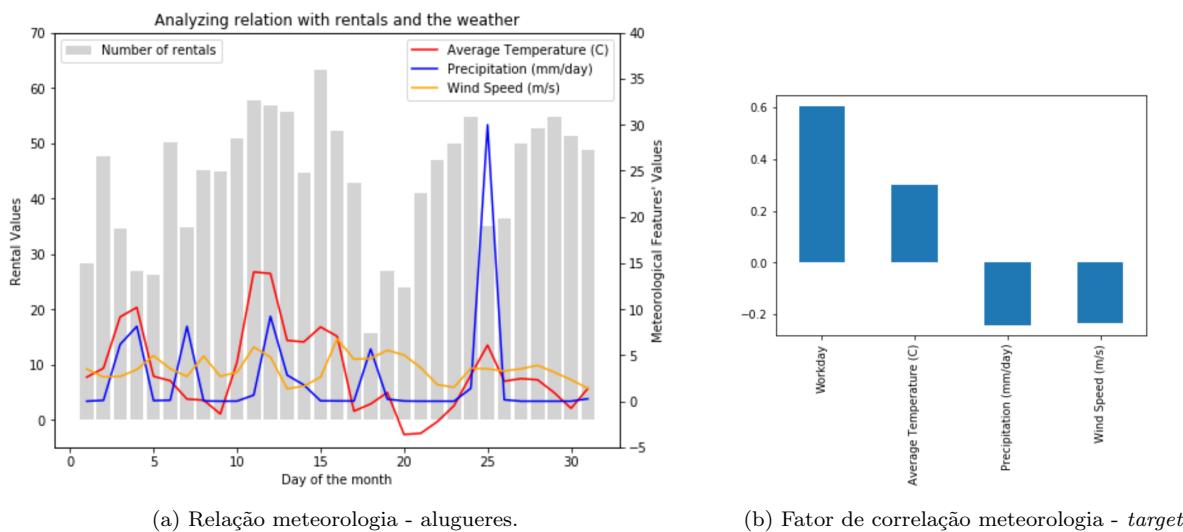


Figure 12: Análise das métricas de meteorologia.

	<b>Modelling Algo</b>	<b>1hour-RMSE</b>	<b>1day-RMSE</b>	<b>10days-RMSE</b>	<b>FutureData-RMSE</b>
<b>0</b>	RandomForestRegressor	67.820662	138.292505	51.359062	64.449856
<b>1</b>	AdaBoostRegressor	135.579663	172.439837	158.508290	128.921034
<b>2</b>	BaggingRegressor	65.558283	151.455177	67.904810	72.649161
<b>3</b>	GradientBoostingRegressor	86.549579	136.614185	101.903747	85.980090
<b>4</b>	SVR	271.678811	267.448851	257.778139	269.499850
<b>5</b>	KNeighborsRegressor	139.514965	168.576271	190.551620	140.473181

(a) Comparação de RMSE nos diferentes modelos.

	<b>Modelling Algo</b>	<b>1hour-R<sup>2</sup></b>	<b>1day-R<sup>2</sup></b>	<b>10days-R<sup>2</sup></b>	<b>FutureData-R<sup>2</sup></b>
<b>0</b>	RandomForestRegressor	0.910736	0.589996	0.952374	0.921700
<b>1</b>	AdaBoostRegressor	0.440174	0.119480	0.411345	0.594810
<b>2</b>	BaggingRegressor	0.923086	0.515432	0.913413	0.897334
<b>3</b>	GradientBoostingRegressor	0.836615	0.580451	0.766438	0.837390
<b>4</b>	SVR	-130.959908	-86.009900	-44.447247	-72.550218
<b>5</b>	KNeighborsRegressor	0.488372	0.207822	-0.324100	0.470675

(b) Comparação de R<sup>2</sup> nos diferentes modelos

Figure 13: Comparação de modelos em termos de RMSE e R<sup>2</sup>.

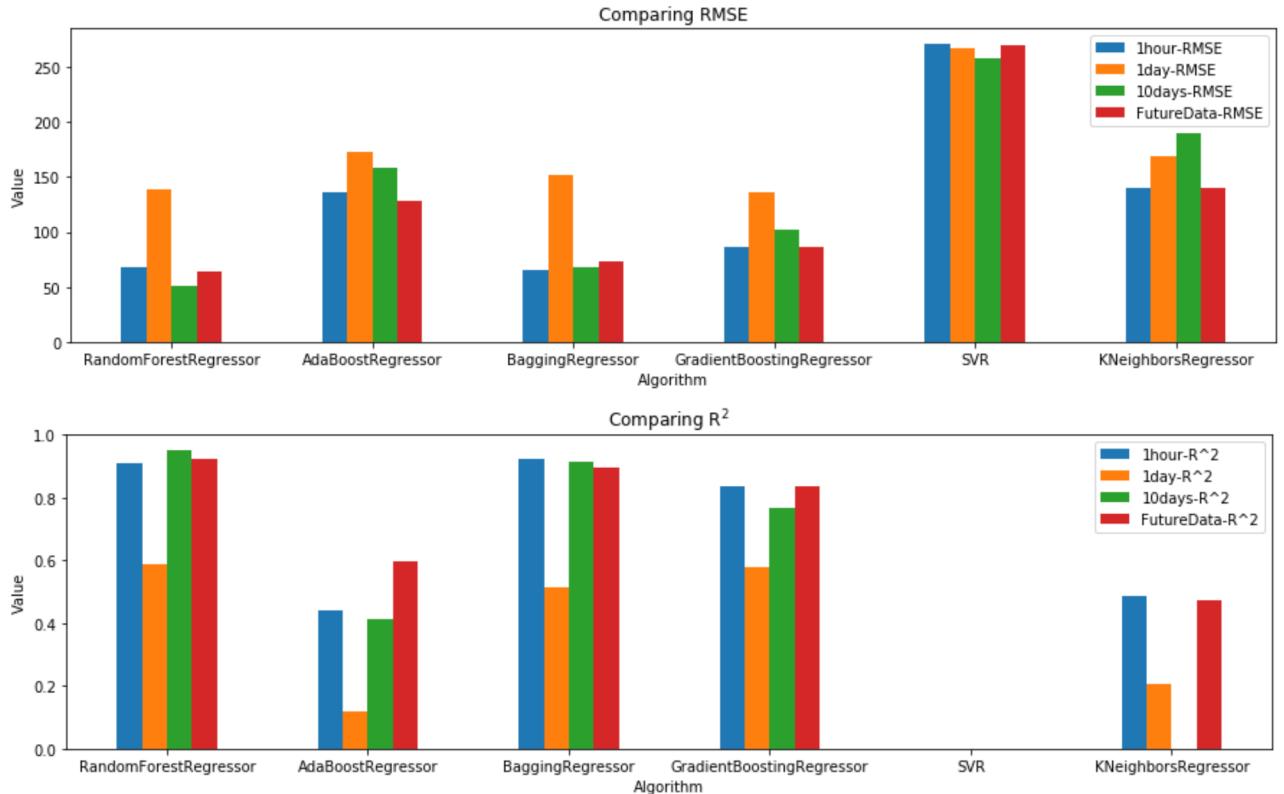


Figure 14: Comparação de modelos em termos de RMSE e R<sup>2</sup>.

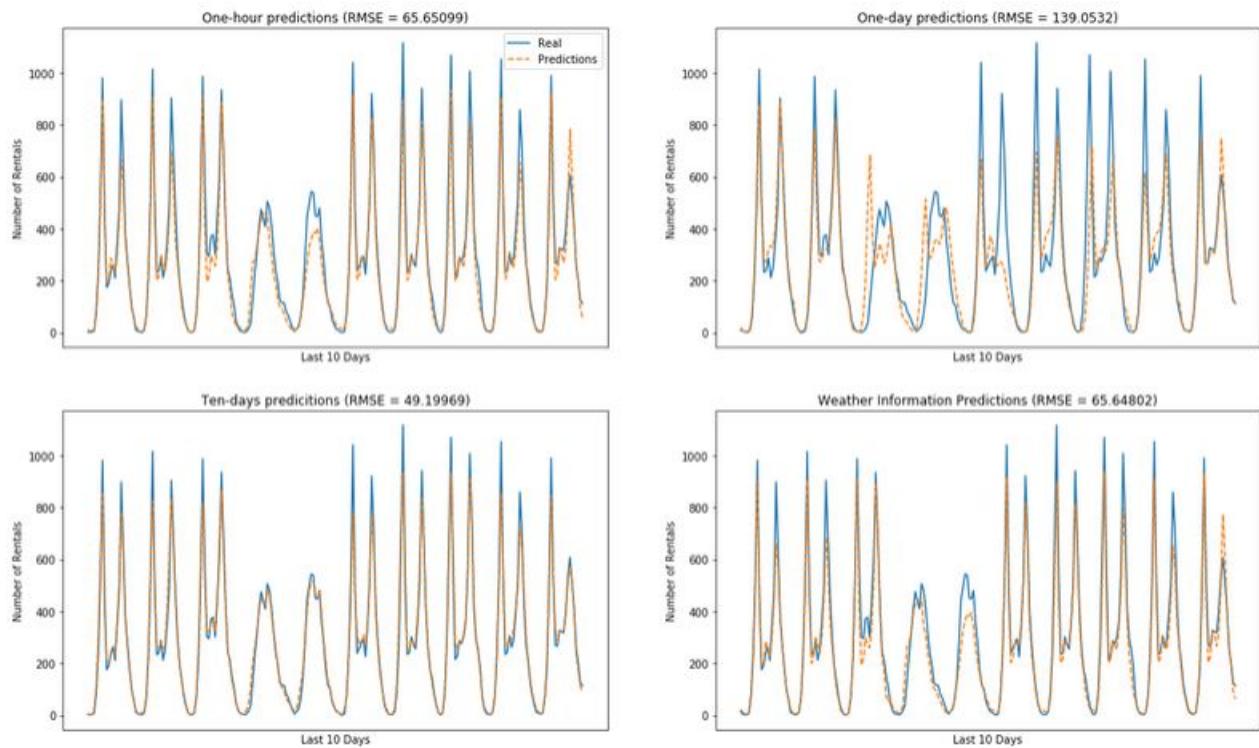


Figure 15: Diferentes previsões através do RandomForestRegressor.