Trabalho 1

INF-0615 – Aprendizado de Máquina Supervisionado I

Grupo:
Evandro Santos Rocha
Laíssa Pacheco de Oliveira
Rafael Dantas de Moura

O objetivo deste trabalho é desenvolver modelos de regressão linear para predizer a concentração de monóxido de carbono no ar.

1. Inspecionem os dados. Quantos exemplos vocês tem?

Com o comando `dim` aplicado aos conjuntos de treino e validação, obtivemos a quantidade de linhas (exemplos) e colunas :

Conjunto	Número de exemplos (linhas)	Número de colunas
Treino	244.582	17
Validação	61.147	17

Já com o comando `summary`, podemos observar cada atributo de cada conjunto. De imediato, podemos observar que nenhum dos conjuntos possui NA para nenhuma coluna.

Observamos também que colunas relacionadas ao dia e hora (year, month, day e hour) possuem valores aparentemente balanceados, pois: as médias e medianas estão próximas da metade entre os mínimos e máximos; e os quartis estão como esperados.

Adicionalmente, verificamos que os conjuntos de treinamento, de validação e (mais tarde) de teste são disjuntos, ou seja, não registro (exemplo) repetido entre os conjuntos. Isso foi feito com o comando `merge`.

Observando as colunas, verificamos que quinze são atributos, uma é o índice do exemplo (variável “no”) e uma é a concentração de monóxido de carbono (variável “target”), totalizando 17.

Como vocês irão lidar com as features (atributos) discretas, se houverem?

De imediato, 5 features são candidatas a discretas: year, month, day, hour e wd (ano, mês, dia, hora, e direção do vento).

As features ano, mês, dia e hora podem ser tratadas como variáveis discretas ou contínuas, dependendo do contexto. Neste exercício, trataremos como contínuas e, portanto, serão utilizadas nas fórmulas das regressões lineares.

Já a feature direção do vento, certamente é uma variável categórica e necessita ser transformada para valor. Para este atributo (variável wd), foram encontrados 16 valores diferentes: E, ENE, ESE, N, NE, NNE, NNW, NW, S, SE, SSE, SSW, SW, W, WNW e WSW. Então, utilizando a técnica de "One-Hot-Encoding", criamos 16 novos atributos (wd_e, wd_ene e assim por diante), um para cada valor, e, no final, removemos o atributo original (wd).

Com isso, dos 15 atributos iniciais, como 1 foi removido e 16 foram criados, os nossos conjuntos passaram a ter 30 atributos.

Há exemplos com features sem anotações? Como vocês lidariam com isso?

Fizemos a verificação de features sem anotações e não foi encontrada nenhuma. O próprio `summary` já tinha nos indicado isso, mas o comando `any(is.na(<dataset>))` nos retornou FALSE para os conjuntos disponibilizados, indicando que não existe NA para nenhum atributo.

Caso houvesse, o tratamento dependerá da quantidade de atributos, por linha, com valor igual a NA. Caso toda a linha não possuísse nenhum valor, a linha seria excluída do conjunto. Se, por outro lado, apenas alguns valores da linha estivessem omitidos, faríamos um tratamento para decidir substituir o NA pela média, mediana, moda, etc. A escolha dependerá do contexto.

2. Apliquem alguma técnica de normalização de forma a deixar os dados mais bem preparados para o treinamento (Min-Max, Z-Norma, etc).

Aplicamos a técnica de normalização Min-max para as colunas: year, month, day, hour, PM2.5, PM10, SO2, NO2, O3, TEMP, PRES, DEWP, RAIN e WSPM.

3. Como baseline, treinem uma regressão linear utilizando todas as features para predizer a concentração de Monóxido de Carbono no ar. Reportem o erro nos conjuntos de treinamento, validação e teste.

Conjunto	MAE Mean Absolute Error	MSE Mean Squared Error	R2 Coeficiente de determinação
Treinamento	370.0937	354861.5	0.7338512
Validação	371.0654	366334.6	0.7262772
Teste	372.3734	362478.3	0.7330248

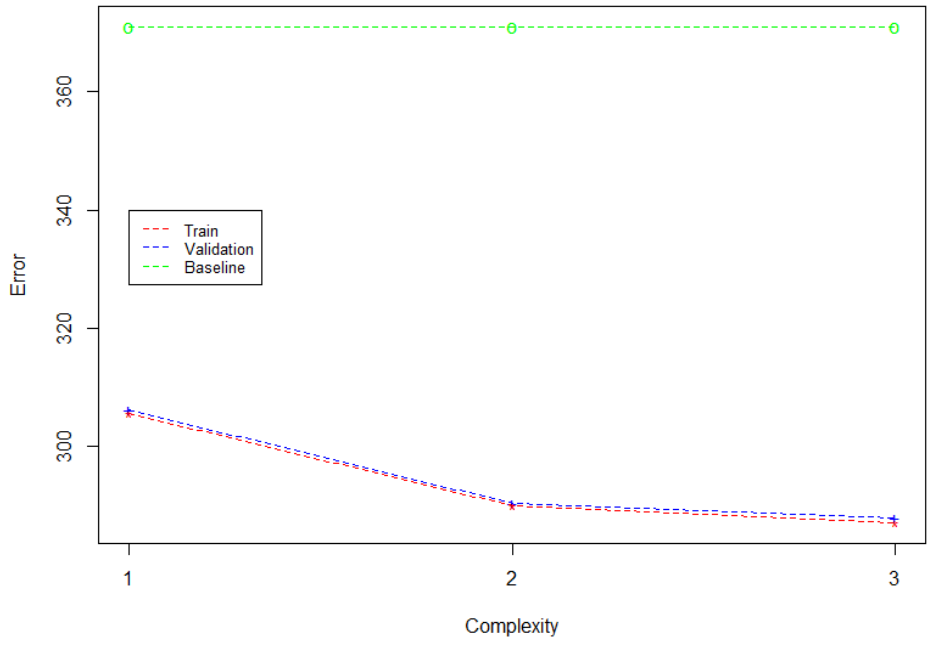
4. Implementem soluções alternativas baseadas em regressão linear através da combinação das features existentes para melhorar o resultado do baseline. Comparem suas soluções reportando os erros no conjunto validação. Tomem apenas a melhor solução baseada no conjunto de validação e reportem o erro no conjunto de teste.

Criamos 3 regressões lineares, combinando as features. Tentamos criar uma quarta, mas a máquina que possuímos não tinha memória suficiente para processar.

Para determinar a melhor solução, dentre as três, olhamos os valores dos erros e plotamos no gráfico. Até f02, os erros entre treinamento e validação decrescem e juntos. De f02 para f03, os erros continuam a decrescer, ainda que em menor intensidade, mas começam a se distanciar. Como não conseguimos fazer uma f04, por questão de recursos, escolhemos a f03 como a melhor e calculamos o erro para o conjunto de teste.

A seguir, estão a tabela com os erros calculados e o gráfico correspondente:

MSE	Treinamento	Validação	Teste
f01	305.6576	306.1869	
f02	290.0147	290.4062	
f03	287.0710	**287.8662**	289.1319



5. Implementem soluções alternativas baseadas em regressão linear aumentando os graus das features (regressão com polinômios) para melhorar o resultado obtido no baseline. Plotem o erro no conjunto de treinamento e validação pelo grau do polinômio. Identifiquem as regiões de underfitting, ponto ótimo e overfitting. Tomem apenas o melhor modelo polinomial baseado no conjunto de validação e reportem seu erro no conjunto de teste.

Criamos 8 regressões lineares com polinômios, onde o de grau 1 é a própria baseline.

Para determinar a melhor regressão, calculamos os erros nos conjuntos de testes e validação e plotamos um gráfico com esses erros. O gráfico nos mostra que os erros decrescem juntos até a 5a regressão. A partir daí, o erro no conjunto

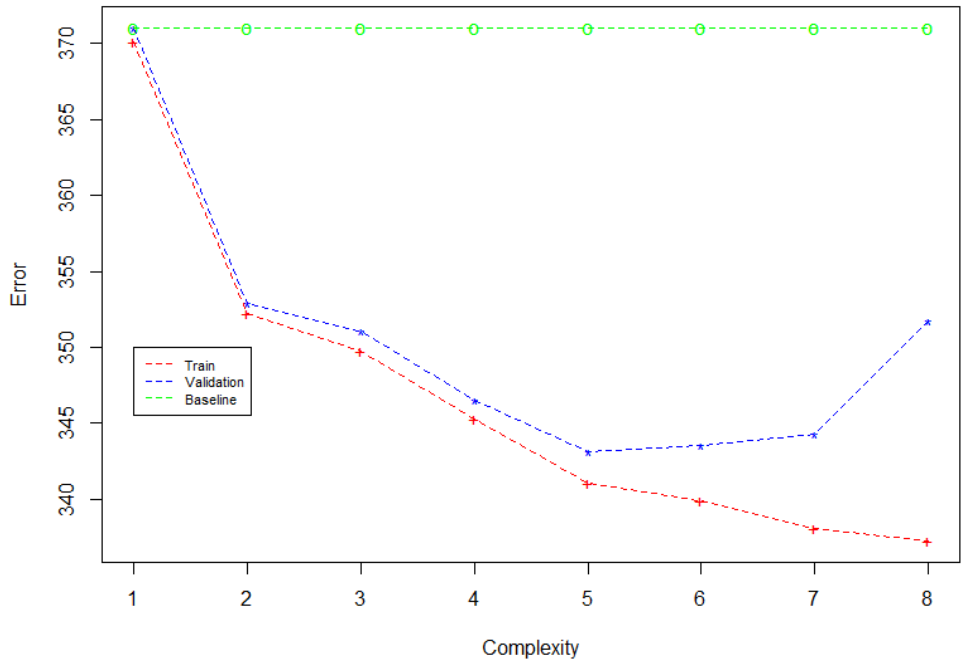
de treinamento continua a decrescer, porém aumenta no conjunto de validação. Isso indica que, a partir da 6a regressão, há overfitting.

Então, o ponto ótimo é retornado pela 5a regressão, o **overfitting** se dá a partir da 6a, e podemos considerar a 1a, baseline, como uma regressão com **underfitting**.

Considerando a 5a regressão como a melhor, calculamos com ela o erro médio absoluto para o conjunto de teste.

A seguir, estão a tabela com os erros calculados e o gráfico correspondente:

MSE	Treinamento	Validação	Teste
Polinômio de grau 1	370.0937	371.0654	
Polinômio de grau 2	352.1933	352.9237	
Polinômio de grau 3	349.7766	351.0492	
Polinômio de grau 4	345.2659	346.4808	
Polinômio de grau 5	341.0588	**343.0935**	344.157
Polinômio de grau 6	339.8772	343.4974	
Polinômio de grau 7	338.0388	344.2929	
Polinômio de grau 8	337.2245	351.757	



6. Escrevam um relatório de no máximo 5 páginas:

(a) Descrevam o que foi feito, bem como as diferenças entre o seu melhor modelo e o seu baseline;

Alguns detalhes estão melhor explicados nas questões acima.

Inicialmente, questão 3, foi realizada uma regressão linear, chamada de baseline, com os dados de treino utilizando todos os atributos, discretos e contínuos. Com essa baseline, foi feita uma predição, onde o erro médio absoluto (MAE) foi de 370,0937 para os dados de treino e de 371,0654 para os de validação.

Em seguida, questão 4, foram realizadas 3 regressões lineares com combinações diferentes de atributos. A melhor regressão gerou erros (MAE) de 287.071, 287.8662 e 289.1319 para os conjuntos de treinamento, validação e testes, respectivamente.

Depois disso, questão 5, foram realizadas regressões com polinômios, gerando até o grau 8. A melhor regressão nesse caso foi a de grau 5, gerando erros (MAE) de 341.0588, 343.0935 e 344.157 nos conjuntos de treinamento, validação e teste, respectivamente. No gráfico plotado para esses erros, é possível observar que do grau 6 em diante o erro de validação aumenta, indicando que, a partir daí, a regressão está gerando **overfitting**.

(b) Reportem o erro do melhor modelo de todos no conjunto de teste. Lembrem-se que o melhor modelo de todos deve ser escolhido baseado no erro no conjunto de validação.

Ao final, o melhor modelo obtido foi o 3° utilizando combinação de atributos (combinação de 4 a 4), pois o erro no conjunto de validação foi mínimo, 287,8662.

Para esta regressão, o erro calculado no conjunto de teste foi de **289,1319**.

(c) Uma Seção de conclusão do relatório explicando a diferença entre os modelos e o porquê que estas diferenças levaram a resultados piores ou melhores.

Como conclusão, ao criar novas features a partir da combinação de features ou a partir da sua potência, obtemos regressões com menores erros. E podemos, ainda, aumentar a combinação ou as potências até obtermos um valor ótimo. Sem dúvida, os dois tipos de regressão se mostraram melhores que a baseline, onde as features foram utilizadas em conjunto, uma vez cada.

No exercício, foi observado que as regressões geradas por combinação de features (2 a 2, 3 a 3, 4 a 4) foram melhores. Como desvantagem, foi observado que existe um custo computacional para obter esse tipo de regressão, pois a combinação de features gera uma quantidade muito grande de novas features. Tanto pela demora para executar quanto pela quantidade de memória necessária. Então, devido ao alto custo da combinação de features, as polinomiais podem ser uma boa alternativa.