**FGV - FUNDAÇÃO GETÚLIO VARGAS  
MBA Executivo em Business Analytics e Big Data**

**ALUNOS**

Arthur Salvador Rocha  
Guilherme Nakano Nalin  
Lucas Gabriel Mendes de Oliveira

Patricia Yoshie Kishi Bueno

Paulo Gustavo Maciel Lopes

**Projeto em Grupo – Modelo de Score de Satisfação de Clientes de uma Companhia Aérea**

Trabalho apresentado para obtenção de nota da disciplina Análise Preditiva  
Professor Dr. João Raphael Dias Pinto

São Paulo/SP  
Unidade Paulista  
Outubro de 2020

**1. Introdução**

Trabalho em grupo realizado para avaliação dos conhecimentos adquiridos na disciplina de Análise Preditiva utilizando a linguagem *R* para análise e construção de modelos preditivos supervisionados.

**2. O Problema**

A satisfação dos clientes é uma das medidas de qualificação de produtos, serviços e recursos de uma empresa que pode evitar problemas mais graves, como a imagem da empresa ou produto prejudicada.

Mensurar a satisfação é um ponto muito importante na jornada do cliente para qualquer tipo de empresa e a modelagem preditiva pode fornecer *insights* para a melhoria da satisfação com técnicas de previsão baseada em interações passadas.

O problema a ser estudado visa a construção de um modelo de *score* de satisfação baseado em informações de passageiros e do voo de uma empresa aérea americana, além de um *survey* enviado a cada passageiro após o voo realizado.

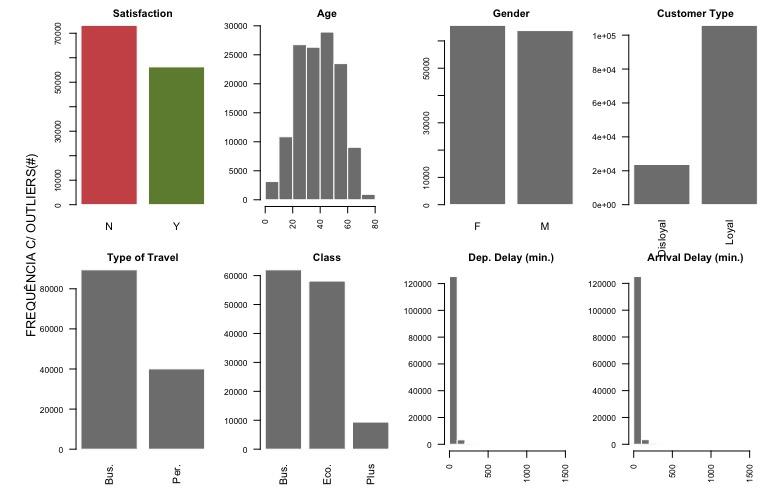
**3. Análise e Tratamento dos Dados**

Foi fornecido um conjunto de dados com 129.880 observações e 24 variáveis, sendo:

Dados do voo

* *id*: código do registro;
* *Satisfaction*: nível de satisfação (*satisfaction, neutral or dissatisfaction*);
* *Age*: idade do passageiro;
* *Gender*: sexo do passageiro (*Female, Male*);
* *Type of Travel*: objetivo do voo (*Personal Travel, Business Travel*);
* *Class*: classe do voo (*Business, Eco, Eco Plus*);
* *Customer Type*: tipo do cliente (*Loyal customer, disloyal customer*);
* *Flight distance*: distância do voo;
* *Departure Delay in Minutes*: atraso na partida em minutos;
* *Arrival Delay in Minutes*: atraso na chegada em minutos;

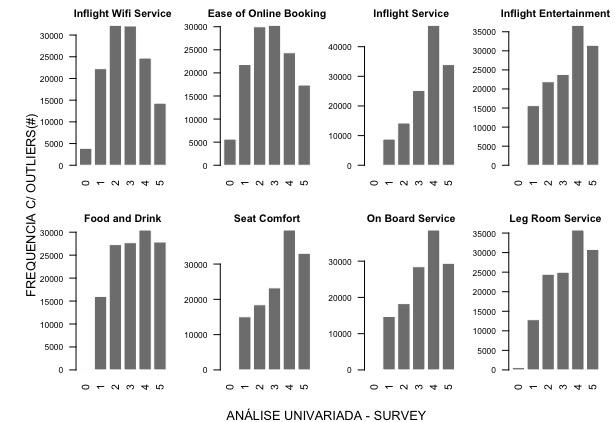
A seguir a distribuição das variáveis da base acima utilizada no trabalho.

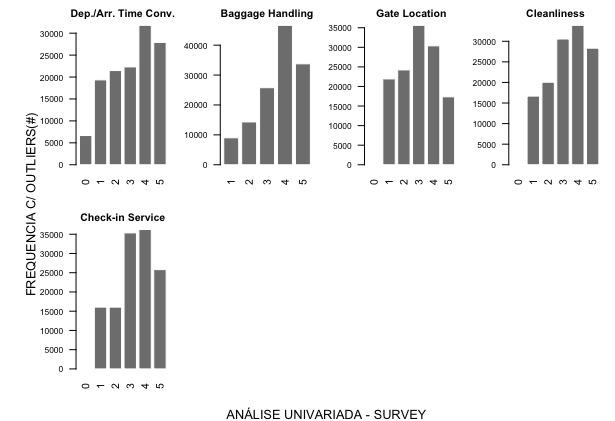
****

Dados do *survey*

* *Inflight wifi service*: nível de satisfação do sinal de *wifi* (0 *Not Applicable*, 1-5);
* *Ease of Online booking*: nível de satisfação realização da reserva;
* *Inflight service*: nível de satisfação do serviço de voo;
* *Inflight entertainment*: nível de satisfação do entretenimento do voo;
* *Food and drink*: nível de satisfação da comida e bebida;
* *Seat comfort*: nível de satisfação do assento;
* *On board service*: nível de satisfação do serviço a bordo;
* *Leg room service*: nível de satisfação do serviço para pernas;
* *Departure/Arrival time convenient*: nível de satisfação do horário de chegada e partida;
* *Baggage handling*: nível de satisfação do manuseio da bagagem;
* *Gate location*: nível de satisfação da localização do portão;
* *Cleanliness*: nível de satisfação de limpeza;
* *Check-in service*: nível de satisfação do serviço de *check-in*;

Abaixo a distribuição das notas do *survey* utilizadas para análise do negócio:





Após o processo de *feature engineering* as variáveis utilizadas no treino dos algoritmos foram tratadas da seguinte forma:

* **Variável Dependente (*target*)**

*Satisfaction* - tratada como variável qualitativa

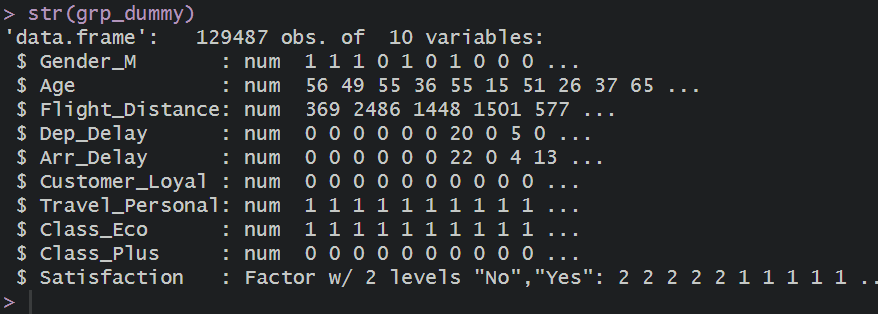
* **Variáveis Independentes (explicativas)**

Qualitativas: *Gender*, *Customer*, *Travel*, *Class*

Quantitativas: *Age*, *Flight\_Distance*, *Dep\_Delay*, *Arr\_Delay*

Assim, a base de dados passou a ser composta por 10 variáveis com 129.487 observações, pois foram identificadas 393 *NAs* na variável *Arrival Delay in Minutes* e retiradas do *data frame*. Para a base de treino foram utilizadas 90.640 observações e na base de teste 38.846 observações.

Abaixo estrutura do *data frame* utilizado na execução dos algoritmos para realização dos treinos e testes da capacidade preditiva de cada modelo:



**4. Modelagem**

Para resolução do problema de negócio, foram construídos quatro modelos distintos de predição:

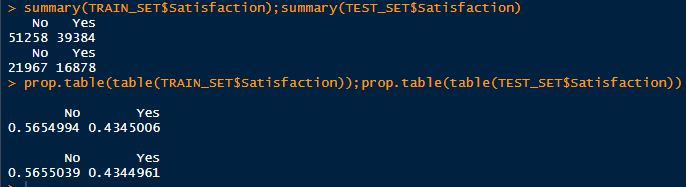
* Modelo de Regressão Logística
* Modelo de Árvore de Decisão
* Modelo de Random Forest
* Modelo de Adaboost

A construção de cada modelo levou em consideração as especificidades e limitações de cada algoritmo, aplicando-se as respectivas técnicas de *tuning* (ajustes finos feitos nos parâmetros que definem os modelos) e para o Modelo de AdaBoost, foi necessário amostrar a base por questões de tempo de execução.

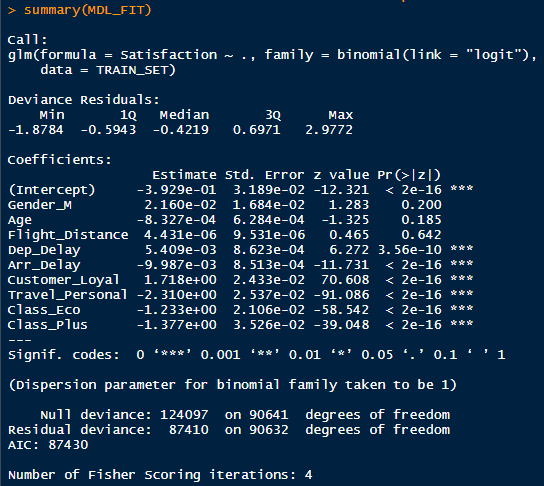
Os quatro modelos foram comparados através das métricas de *Gini*, *AUC*, Especificidade e Sensibilidade, e o vencedor foi escolhido para resolução do problema de negócio.

**4.1 Regressão Logística**

A partir da base tratada e das variáveis quantitativas devidamente convertidas em *dummies*, realizamos a divisão dos conjuntos de treino e teste numa proporção de 70% e 30% respectivamente e conferimos a distribuição da variável resposta nestes conjuntos. Visto a característica da regressão logística não analisaremos a distribuição das variáveis explicativas neste caso.

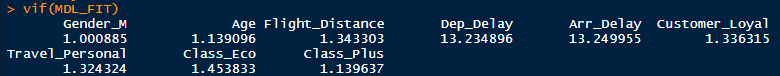


Obtendo uma distribuição equilibrada da variável *Satisfaction* entre os conjuntos, seguimos para o treino inicial da regressão logística binomial, considerando todas as variáveis explicativas, a partir do qual faremos a análise dos coeficientes:

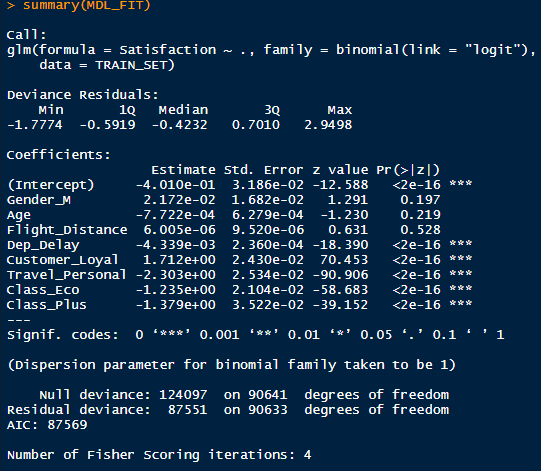


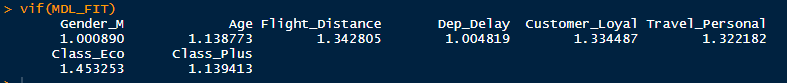
Podemos constatar que as variáveis *Gender*, *Age* e *Flight\_Distance* tem uma significância estatística muito baixa quando observamos o p-valor, o que impacta na obtenção de um AIC (*Akaike Information Criterion*) de 87430 obtido neste teste.

Verificaremos também a multicolinearidade entre as variáveis de acordo com a métrica VIF (*Variance Inflation Factor*), indicando o quanto um coeficiente da regressão é inflado devido à multicolinearidade.



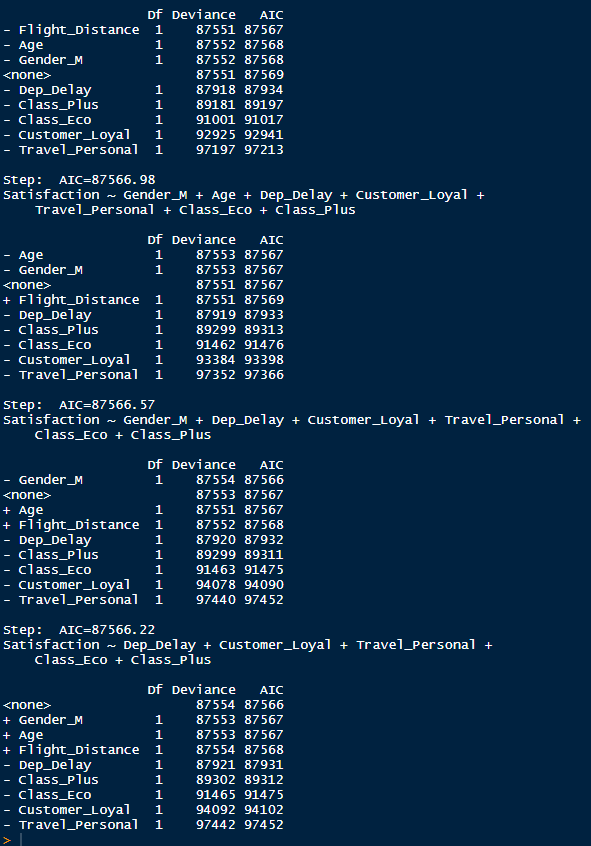
Observamos que as variáveis *Dep\_Delay* e *Arr\_Delay* se tem multicolinearidade, apresentando um VIF alto, acima de 10, de maneira que podemos remover uma delas do modelo. Visto que o atraso na chegaram é impactado pelo atraso na decolagem, tendemos a remover a variável *Arr\_Delay*, ficando com a seguinte redistribuição dos coeficientes e VIF:



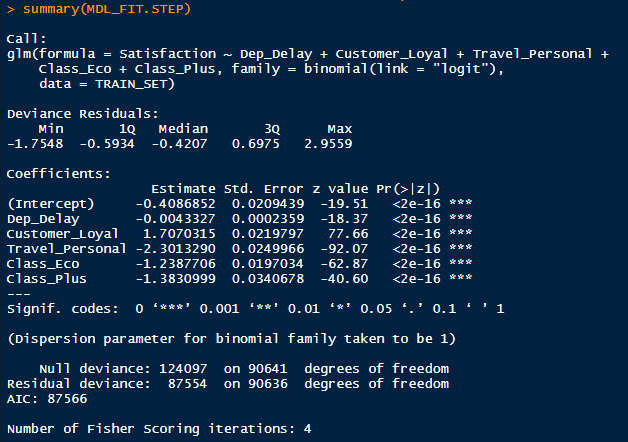


Observando o VIF podemos constatar que não temos nenhum outro caso de multicolinearidade. No entanto, tivemos um leve aumento no AIC, que se explica pela remoção de uma variável com significância estatística sobre o modelo.

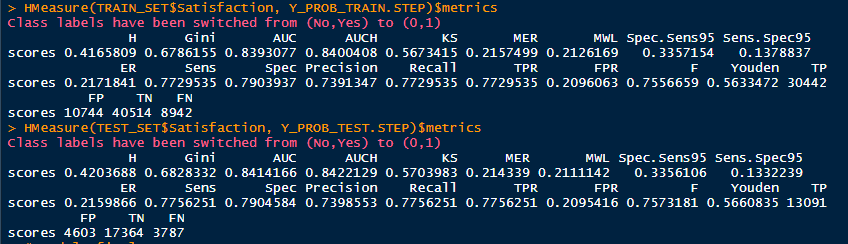
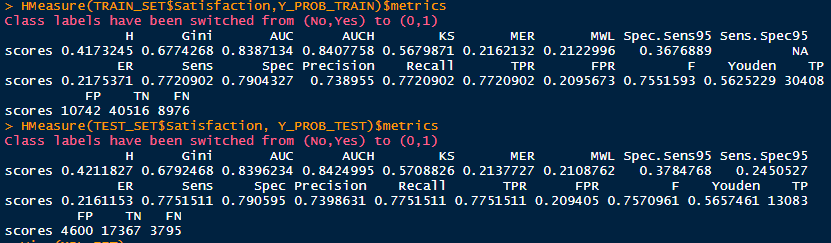
Na sequência aplicamos o processo *Stepwise Regression* para encontrar o conjunto com menos variáveis e maior poder de explicação da variável resposta, detalhando as variáveis removidas em cada etapa:



Foram executadas quatro etapas no processo, removendo em cada uma delas sequencialmente as variáveis: *Flight\_Distance*, *Age* e *Gender\_M* resultando em um AIC de 87566, uma redução de aproximadamente 4 pontos quando ao AIC inicial, demonstrando que houve melhora nos coeficientes do conjunto após a remoção das variáveis, que também eram as com menor significância estatística.

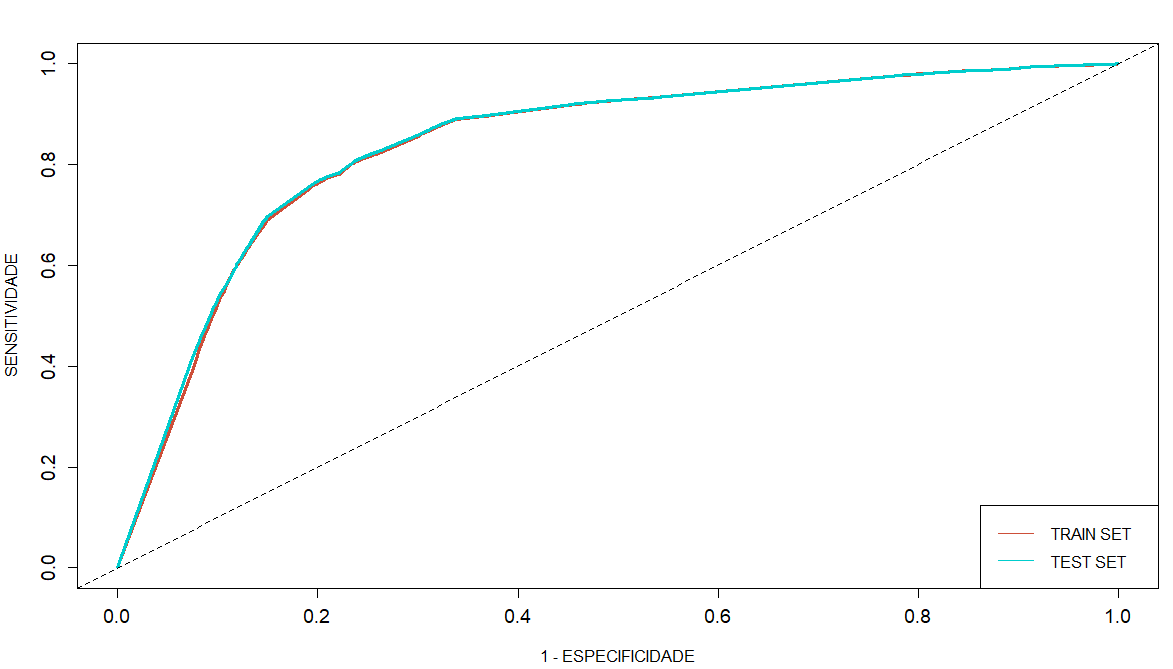
****

Com o modelo ajustado, seguiremos realizando as predições no modelo completo e no modelo com *Stepwise* aplicado e posterior comparação das métricas de desempenho para verificar qual deles tem melhor poder de predição.



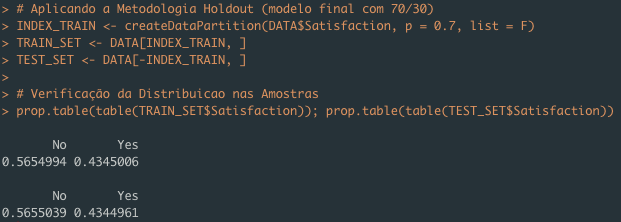
Avaliando os valores de Gini e AUC podemos observar que ambos os modelos apresentaram boa capacidade de generalização e predição, tendo variado pouco entre as predições de treino e testes com valores de Gini aproximadamente de 68% e AUC de 83%. Porém, nas predições do modelo com *Stepwise* observamos um pequeno ganho em ambas as métricas, de maneira que entre os dois modelos consideramos este como o melhor.

A seguir apresentamos a curva ROC do modelo com *Stepwise*, comprovando visualmente o alto grau de generalização do modelo.

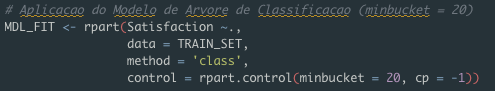


**4.2 Árvore de Decisão**

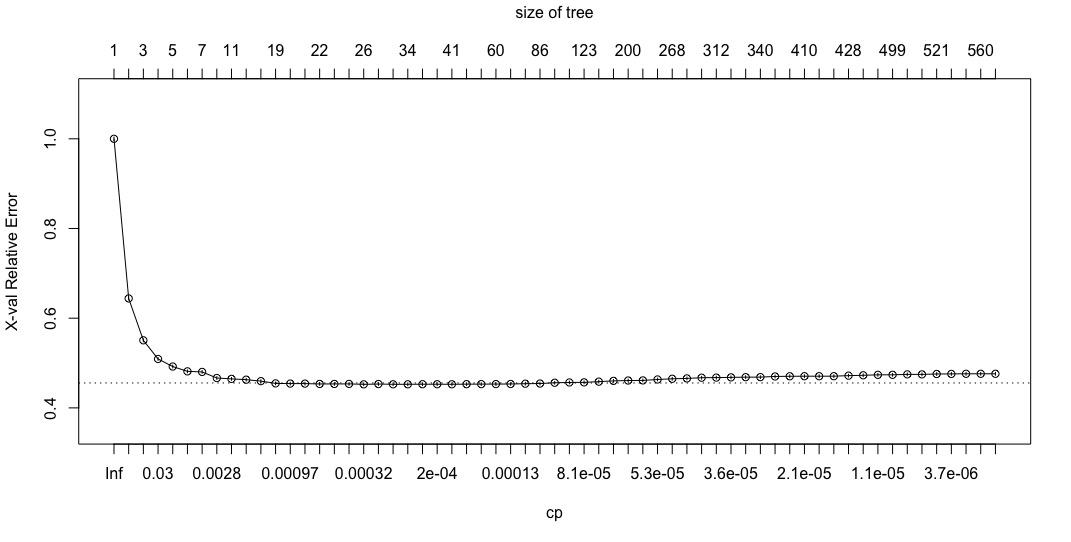
A partir da base com as variáveis convertidas em *dummies*, foi realizado o procedimento *holdout* onde foi aplicado um particionamento de 70% dos registros para o treinamento do modelo e os demais 30% para a realização dos testes. Após isso, foi declarado o método de *summary* para garantir que as bases possuem métricas semelhantes entre si:



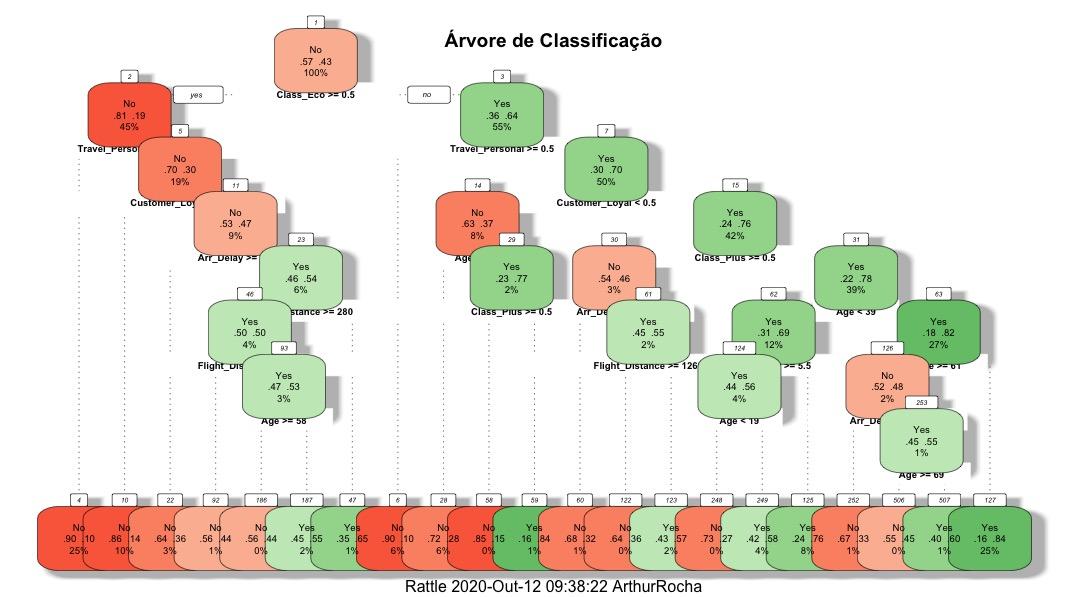
Com essa etapa confirmada, foi aplicado a criação da árvore de classificação com os seguintes parâmetros:



* Variável target '***Satisfaction***’ em função de todas as variáveis presentes na base;
* *method* - Tipo de modelo para Classificação ‘*class*';
* *minbucket* - Tamanho mínimo de 20 registros nos nós terminais;
* *cp* - Parâmetro de definição para desenvolvimento da árvore até o limite da redução do erro (-1).

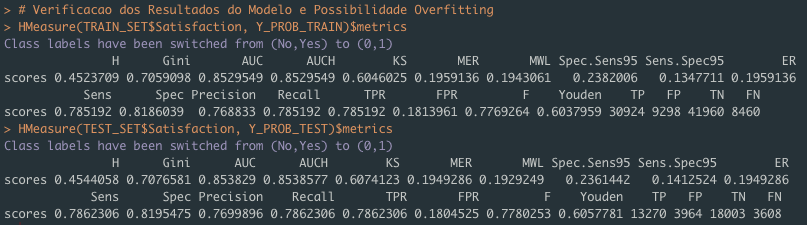
Com o modelo de treino criado, pôde ser iniciada a etapa de poda da árvore de classificação, simplificando-a e minimizando a possibilidade de *overfitting*. Para isso, foi plotado o gráfico com a redução do erro de acordo com a extensão da árvore, possibilitando a análise em qual ponto não há uma redução significativa desse erro. Com a localização desse número aproximado, foi realizada a poda com o número de *cp* de 7.1095e-04 (localizado na profundidade de número 14 da árvore apresentada no gráfico abaixo):  


Com essa etapa concluída, é possível a visualização do modelo final da árvore de classificação:

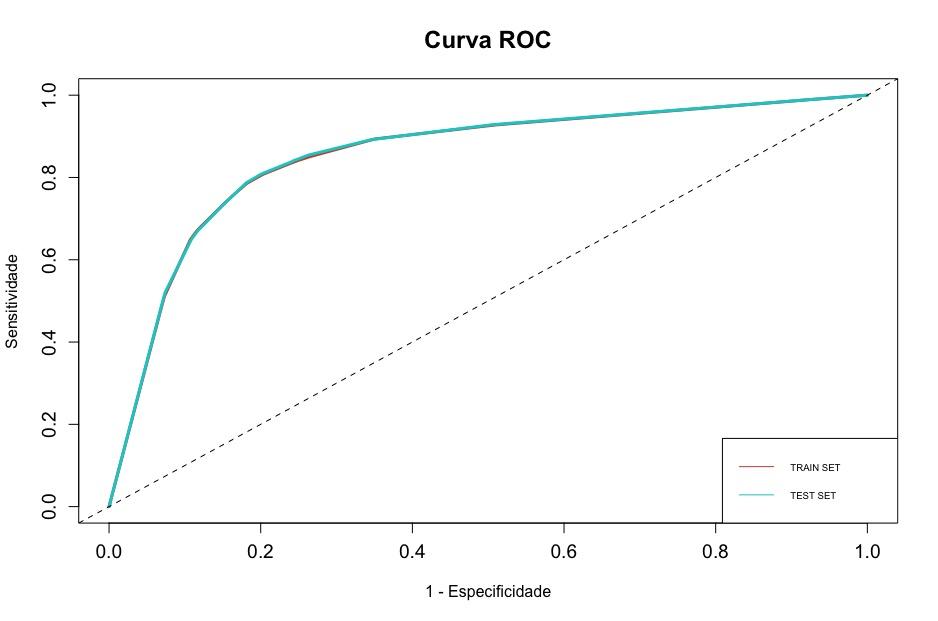


A partir da conclusão da etapa de testes do modelo de treino, é possível a aplicação do modelo à base de teste para a apuração da possibilidade de *overfitting* no modelo e na sua eficiência preditiva:  


Para a realização da conferência de *overfitting,* foi comparado os indicadores de AUC e Gini das duas amostras. Como podemos ver no quadro abaixo, tanto o AUC (Treino = 0.8530 x Teste = 0.8538) quanto o Gini (Treino = 0.7059 x Teste = 0.7077) apresentaram resultados semelhantes, garantindo dessa forma a ausência de *overfitting* e garantindo o poder de generalização do modelo:



Com a demonstração da Curva ROC, podemos verificar que a performance entre as duas amostras é semelhante, garantindo a sobreposição das duas linhas praticamente durante toda a extensão do gráfico:

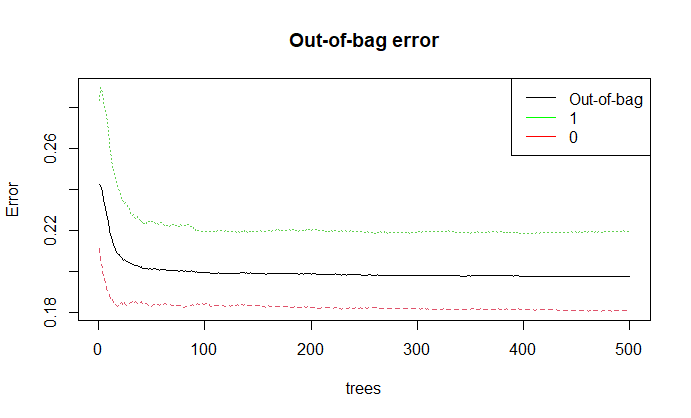


**4.3 *Random Forest***

O algoritmo *Random Forest* foi configurado através de três parâmetros:

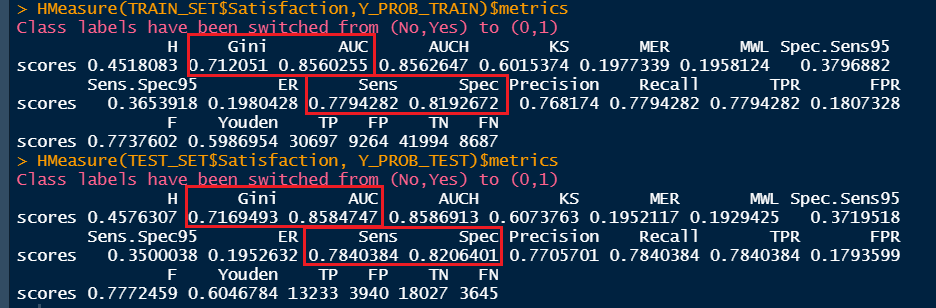
* *mtry* - número de variáveis escolhidas aleatoriamente a cada divisão
* *nodesize* - tamanho mínimo dos nós terminais
* *ntree* - quantidade de árvores a serem construídas pelo algoritmo

O primeiro passo adotado foi a definição do ponto de poda, no caso, o número mínimo de árvores a partir do qual não existe diminuição significativa nos valores de erro. Para isso o modelo foi executado com os valores padrões vistos em sala de aula (*mtry* = 4 / *nodesize* = 10 / *ntree* = 500), e em seguida foi gerado a curva de erro “*out of bag*”, mostrada abaixo:



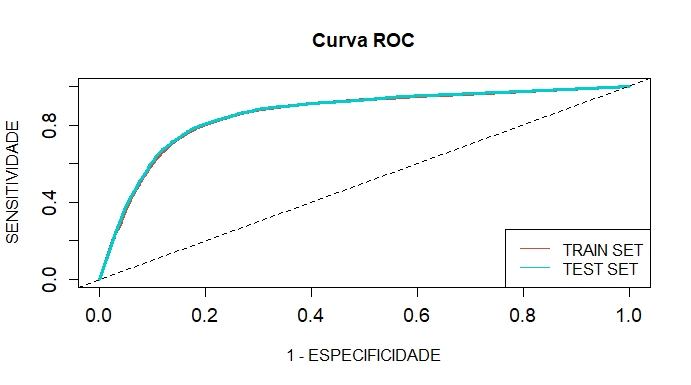
A linha preta representa a curva de “Erro x Quantidade de árvores” do modelo. A partir dela, foi escolhido ponto de poda em **100 árvores (*ntree* = 100)**.

Em seguida, foram feitas várias execuções com diferentes combinações de valores de *mtry* e *nodesize*, cujos resultados foram analisados através do *Gini*, *AUC*, Sensitividade e Especificidade. A melhor performance foi atingida com *mtry* = 4 / *nodesize* = 20 / *ntree* = 100, conforme mostrado na imagem abaixo:



Os quatro valores de cima correspondem a performance do modelo na base de treino, enquanto que os quatro debaixo correspondem a performance na base de teste. Esses valores mostram que o modelo apresentou uma boa capacidade de generalização, pois tanto o *Gini*, como o *AUC* apresentaram pouca variação entre treino e teste. Além disso, no teste, o modelo apresentou desempenho ligeiramente superior para prever *TP* (verdadeiros positivos) e *TN* (verdadeiros negativos), evidenciado acima pelo aumento que as taxas de Especificidade e Sensitividade apresentaram em comparação com o treino.

A curva ROC mostrada abaixo também foi utilizada para avaliação da performance:



A curva *ROC* mostra de forma visual a pequena variação vista nas métricas de *Gini* e *AUC* (entre teste e treino). Como o algoritmo teve boa capacidade de generalização, as curvas são praticamente sobrepostas.

**4.5 AdaBoost**

Devido às limitações desse algoritmo, capacidade de processamento, foi necessário amostrar a base de dados para possibilitar o término de execução do algoritmo e uma possível avaliação do modelo para conclusão desse trabalho. Mesmo utilizando a técnica de processamento paralelo com as bibliotecas *doParallel* e *foreach*.

O algoritmo *AdaBoost* foi configurado através de três parâmetros:

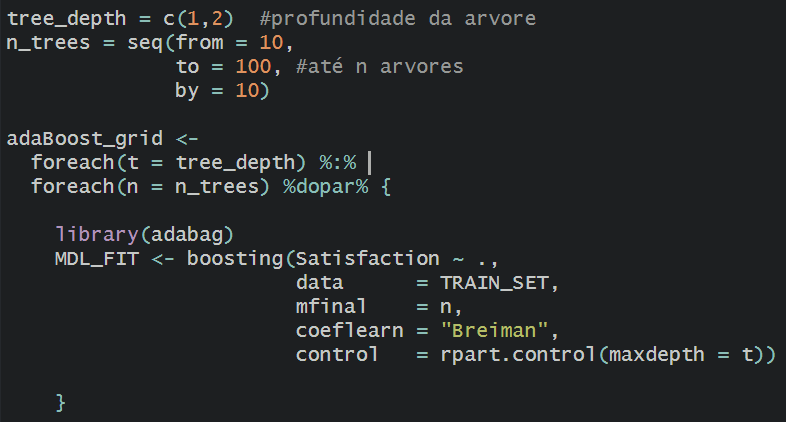
* mfinal- quantidade de árvores
* *maxdepth* - profundidade
* *coeflearn* = *Breiman* (mantido conforme sala de aula devido ao tempo de processamento para demais testes)

Na primeira tentativa de executar o algoritmo com 100% da base para gerar modelos com as profundidades 1,2,3 e até 250 árvores foi abortada após 19 horas de execução.

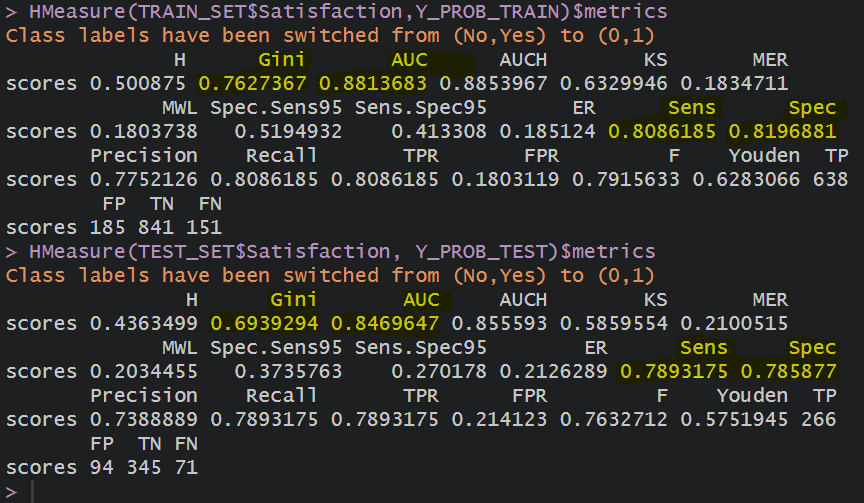
Na segunda tentativa, utilizando a função *createDataPartition()* foi pego uma amostra de 10% da base para gerar modelos com profundidade 1 e 2 e até 250 árvores. Até a entrega desse trabalho, o algoritmo não concluiu sua execução, somando 5 dias e 2 horas.

Uma terceira tentativa em paralelo foi executada com 5 horas de processamento, com uma amostra de 1% da base para gerar modelos com profundidade 1 e 2 e até 50 árvores.

O grupo não satisfeito, atreveu-se a última execução com uma amostra 2% da base para gerar modelos com profundidade 1 e 2 e até 100 árvores. Valores baseados nos outros algoritmos (100 árvores) e tempo de execução da terceira tentativa (1% da base com 5 horas de processamento). O tempo para essa execução foi de 14 horas, parametrizados no algoritmo conforme abaixo:

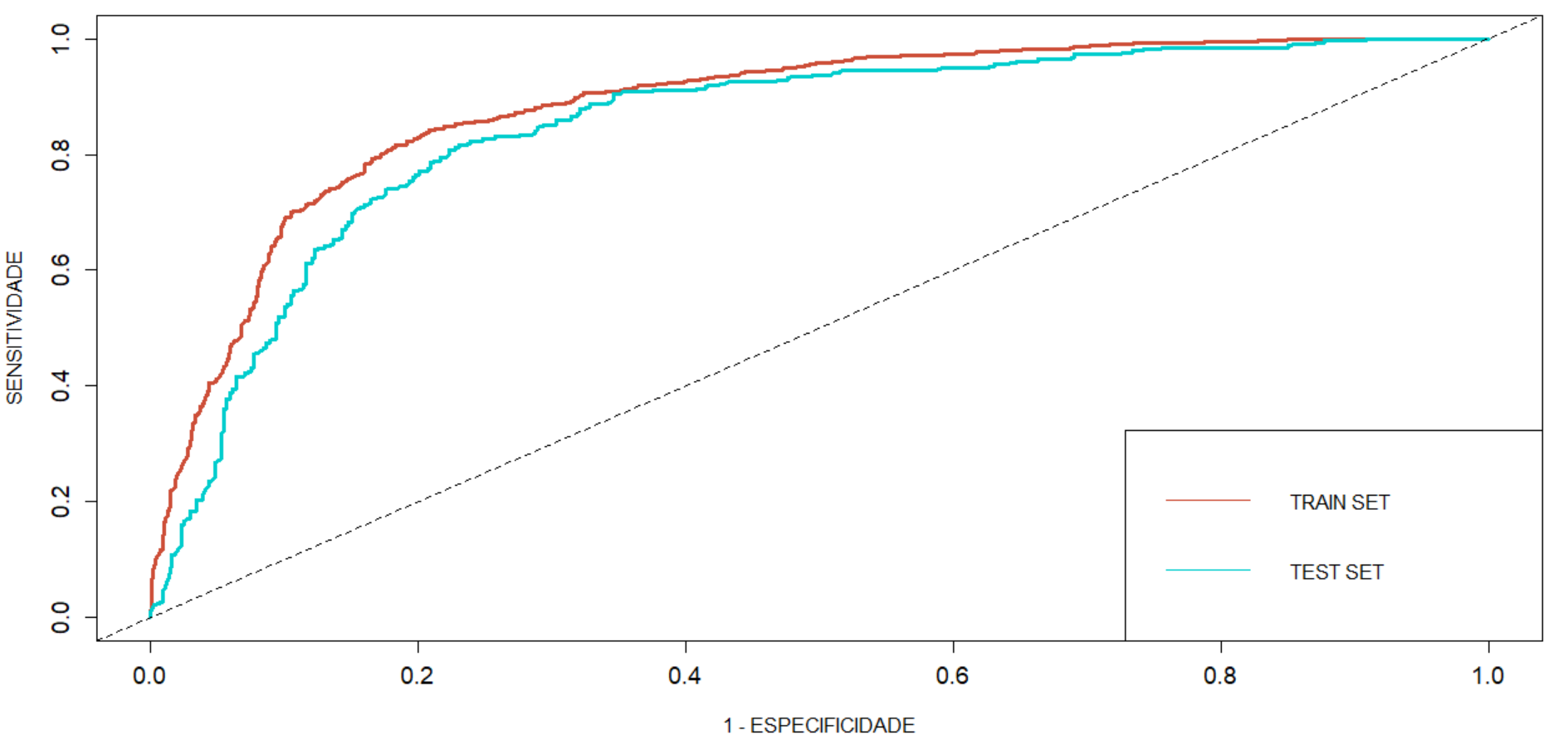


O modelo que apresentou o melhor resultado, dentre as tentativas de processamento realizadas e concluídas foi com profundidade 2 e até 100 árvores, apresentando as métricas abaixo de *Gini*, *AUC*, Sensitividade e Especificidade:



Foi observado que conforme aumentam os valores de *tuning* as métricas também aumentam, assim como a diferença nos valores do *Gini* para todos os modelos gerados, sempre diferem mais do que 5% entre o treino e teste. Podemos também observar que os valores do modelo no teste sempre são muito inferiores, exceto para as taxas de erro.

Com a curva *ROC* abaixo, apresentamos essas diferenças:



Assim, observamos que o modelo não apresenta um bom poder de generalização, apresentando sinais de *overfitting*.

**5. Comparação entre os modelos**

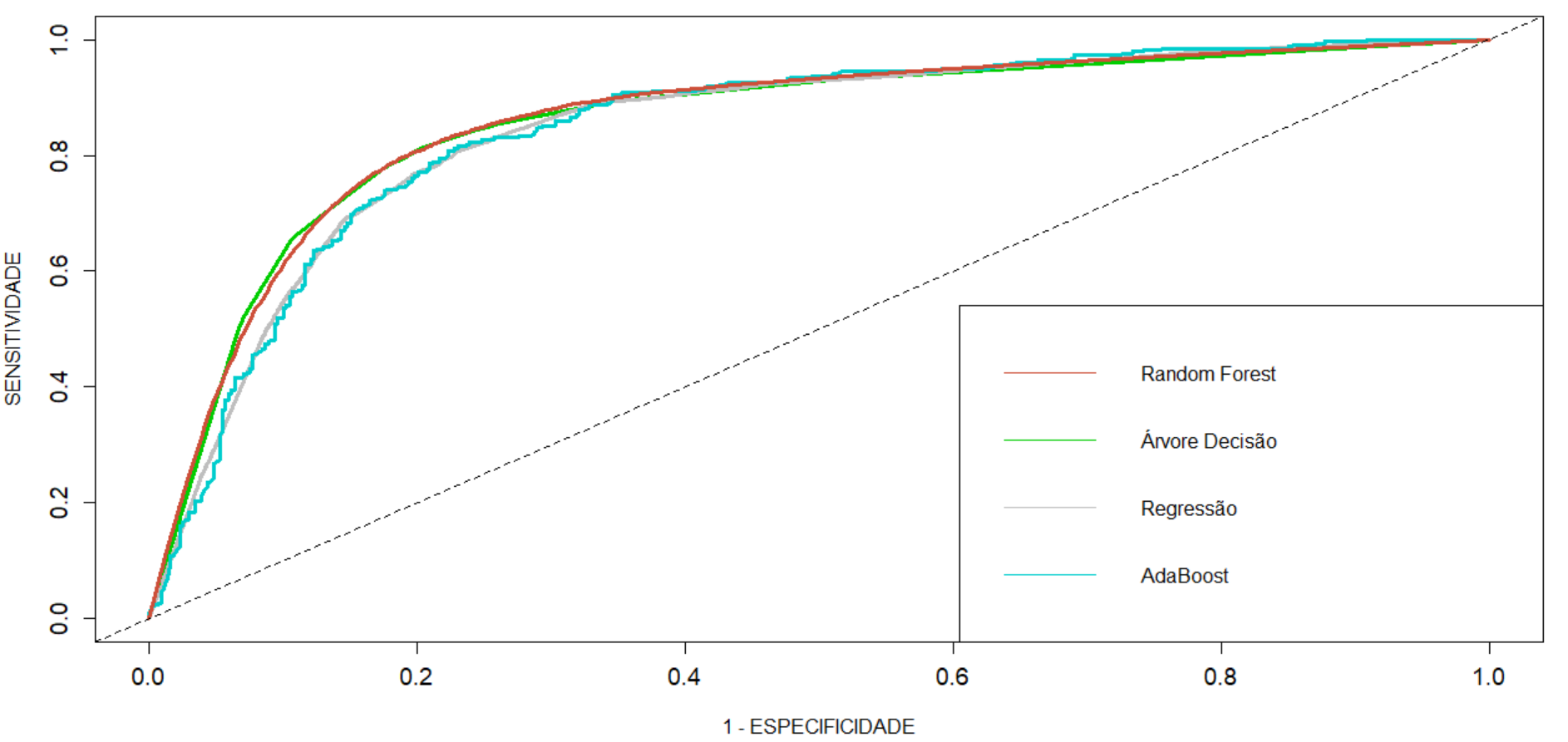
Conforme apresentado os resultados dos respectivos modelos no capítulo anterior, foi possível realizarmos um comparativo de performance dos mesmos a fim de garantir o modelo que nos retornava a melhor predição dos dados.

Para isso, foi montado um comparativo entre eles devido a similaridade dos indicadores para verificação de performance, que são eles: AUC, Gini, Sensitividade, Especificidade e Curva ROC.

Conforme podemos observar no gráfico abaixo, tivemos uma superioridade da modelagem baseada em *Random Forest* na qual teve um desempenho melhor em três dos quatro indicadores de controle utilizados para aferir o melhor modelo, perdendo apenas no quesito de Sensitividade para o algoritmo de *Adaptive Boosting*:

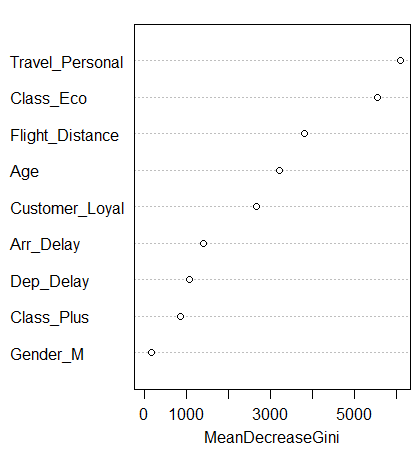


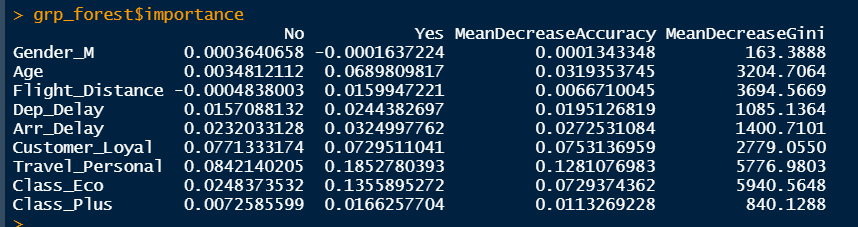
No gráfico abaixo, podemos observar a mesma superioridade de predição do algoritmo de *Random Forest* quando comparamos a performance das Curvas ROC dos respectivos modelos, com desempenho semelhante, mas discreta superioridade em relação a árvore de decisão.



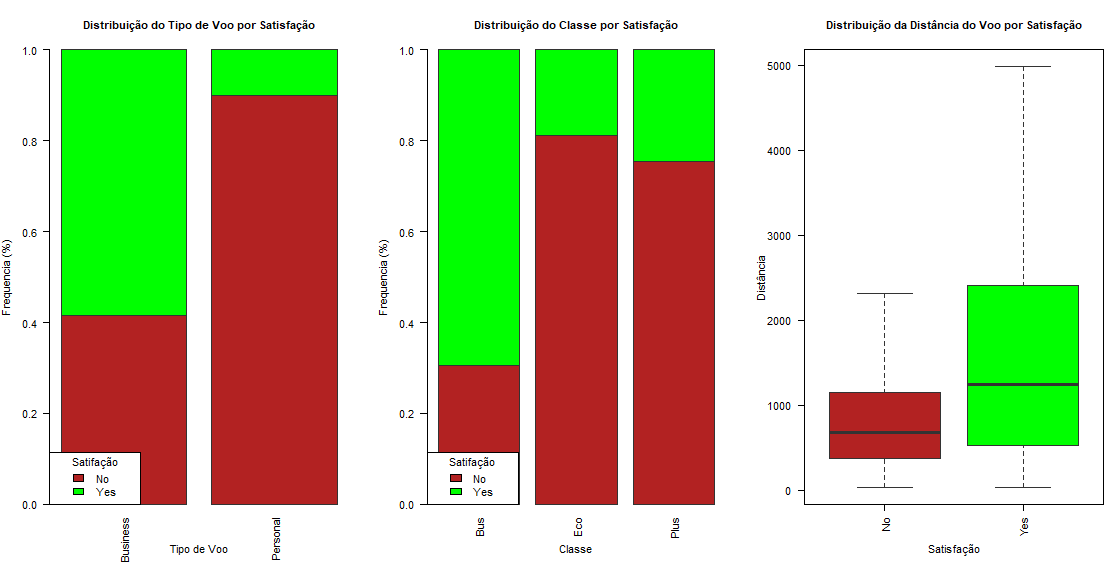
**6. Importância das variáveis por vencedor**

Eleito o modelo de predição com *Random Forest* como vencedor, vamos analisar a importância das variáveis na construção do modelo observando os valores para métrica *MeanDeacreaseGini*, que demonstra o quanto a variável reduz a impureza do nós, ponderando pela média das árvores de decisão na floresta. Com isso observamos que as três variáveis que mais reduzem, em média, a impureza do modelo são respectivamente da maior para menor: Travel\_Personal, Class\_Eco e Flight\_Distance.





Para estas faremos uma análise bivariada, para ver como se distribui a satisfação dos passageiros:

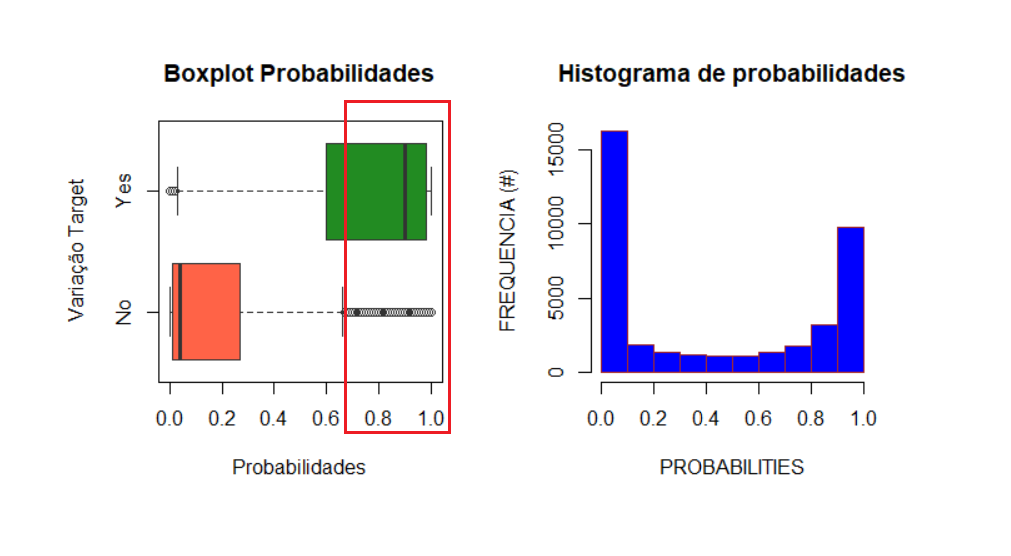


A partir da análise da distribuição da satisfação entre as variáveis de maior importância, concluímos que as viagens com objetivo pessoal, voos nas classe Eco e Plus em curtas distâncias, tendem a ter um maior nível de insatisfação. Observamos também que a distribuição da satisfação nestas variáveis promove uma clara separação das categoria de satisfação, o que corrobora os altos valores de redução da impureza obtidos no *MeanDeacreaseGini.*

**7. Aplicação 1**

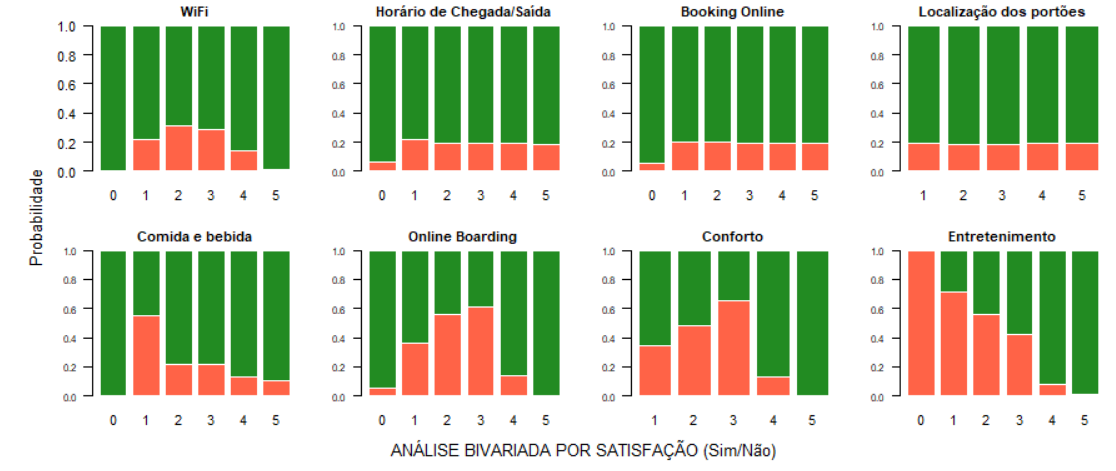
A previsão de satisfação feita pelo modelo construído, é mostrada através de uma probabilidade, que representa a chance de o cliente estar ou não satisfeito.

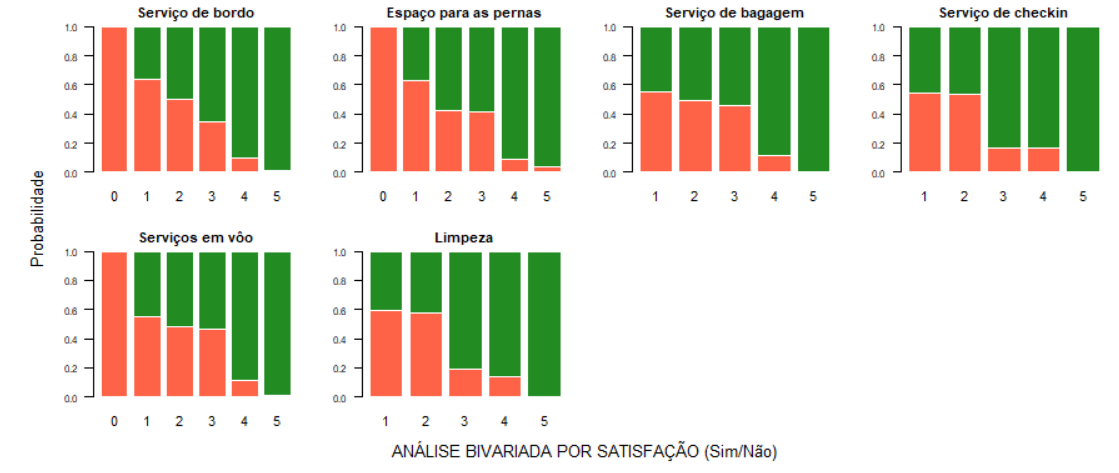
O primeiro pedido de negócios, foi para que fossem separados da base de previsão, os clientes mais propensos a estarem satisfeitos ou não (25% mais de cada). Abaixo vemos a distribuição das probabilidades do modelo:



A faixa dos 25% mais propensos a estarem satisfeitos possui muito mais clientes que a faixa dos 25% mais propensos a não estarem satisfeitos. Ambas as faixas foram separadas da base principal e colocadas em duas novas bases.

A seguir, foram feitas análises bivariadas com todas as variáveis das questões do *survey*, conduzido após o vôo, para essa faixa de clientes:





O eixo x representa a nota atribuída em cada categoria do *survey*, o eixo y expressa a divisão da probabilidade entre os dois grupos. A cor verde representa os clientes mais propensos a estarem satisfeitos, enquanto que a cor vermelha representa aqueles mais propensos a não estarem satisfeitos. Pela análise vemos que algumas categorias têm um maior impacto na insatisfação dos clientes: entretenimento, serviço de bordo, espaço para pernas, serviço de bagagem, serviços em voo, limpeza e conforto. Essas categorias possuem uma proporção muito maior de clientes insatisfeitos atribuindo notas baixas (inferiores a 3), o que indica que podem ser os fatores mais decisivos nos quais a companhia aérea deveria focar os esforços para melhorar.

**8. Aplicação 2**

Com o modelo vencedor (*Random Forest*) foram simulados vários pontos de corte distintos na matriz de confusão:

|  |  |  |  |
| --- | --- | --- | --- |
| *Corte* | *Sensitividade* | *Especificidade* | *Acurácia* |
| *50,00* | *77,91%* | *82,32%* | *80,40%* |
| *40,00* | *80,84%* | *79,75%* | *80,22%* |
| *30,00* | *83,68%* | *76,76%* | *79,77%* |
| *60,00* | *74,67%* | *84,45%* | *80,20%* |
| *70,00* | *69,77%* | *86,83%* | *79,42%* |

Para melhoria dos negócios, é muito usado o contato telefônico precoce de pós-venda com clientes potencialmente insatisfeitos. Sugerimos então que o modelo seja usado para pesquisa do maior número de clientes insatisfeitos.

No nosso modelo o evento pesquisado é "cliente satisfeito". Dado esse contexto, o modelo deve ser capaz de incluir o maior número de 'verdadeiros negativos', ou seja, os clientes que de fato ficaram insatisfeitos, para que um contato precoce possa reverter a percepção do maior número de clientes antes mesmo da resposta da pesquisa de satisfação e dessa forma melhorar a sua percepção e fidelizá-loes.

Foi escolhido o ponto de corte em 70%, o qual trouxe a mais alta Especificidade entre as simulações, ou seja, é o ponto de corte que propiciará o maior número de "verdadeiros negativos", clientes de fato insatisfeitos, para que a ação de pós-venda precoce traga mais benefícios.

**9. Conclusão**

Com a finalização do estudo é possível afirmar que com a base recebida com suas 24 variáveis (8 contando apenas as independentes) e com seus 129.880 registros, foi exequível a criação de uma modelagem preditiva baseada nos algoritmos de Regressão Logística, Árvore de Decisão, Random Forest e AdaBoost, onde foi observado a melhor capacidade de generalização e predição no modelo Random Forest, garantindo um indicador de *Gini* 71% e *AUC* 85% nas bases de treino e teste.

Através desse modelo conseguimos resolver o problema proposto e indicar para a companhia aérea os serviços que requerem mais atenção e investimento para que, de forma preventiva, sejam melhorados para garantir uma melhor percepção dos clientes As categorias foram: entretenimento, serviço de bordo, espaço para pernas, serviço de bagagem, serviços em voo, limpeza e conforto.

Também com o uso do modelo, propomos uma abordagem precoce de pós-venda, com contato precoce com clientes potencialmente insatisfeitos, melhorando assim a sua percepção e aumentando a fidelização.