MBA Business Analytics

Análise de Séries Temporais

Rafael Scopel

Sumário

- Apresentações
- Regras do Jogo
- Introdução as Séries Temporais e Projeção
- Dados no contexto de Séries Temporais
- Avaliação de Performance
- Métodos de Projeção: Visão Geral
- Métodos de Regressão
- Métodos de Projeção: Suavização
- Modelos de Regressão ARIMA

Sumário

- Apresentações
- Regras do Jogo
- Introdução as Séries Temporais e Projeção
- Dados no contexto de Séries Temporais
- Avaliação de Performance
- Métodos de Projeção: Visão Geral
- Métodos de Regressão
- Métodos de Projeção: Suavização
- Modelos de Regressão ARIMA

Apresentações

Prof. MSc. Rafael Scopel

Doutorando em Economia Aplicada pela FGV. Executivo do mercado financeiro, atualmente Líder de MIS na Stone CO. Ex-sócio da PwC na área de Consultoria, atendendo clientes com Banco Itaú-Unibanco, Santander, BB, CEF, BM&F Bovespa entre outros. Participou de projetos de modelagem de risco de crédito e mercado, validação de modelos, análise quantitativa, precificação de produtos de tesouraria e gerenciamento de riscos financeiros.

Sumário

- Apresentações
- Regras do Jogo
- Introdução as Séries Temporais e Projeção
- Dados no contexto de Séries Temporais
- Avaliação de Performance
- Métodos de Projeção: Visão Geral
- Métodos de Regressão
- Métodos de Projeção: Suavização
- Modelos de Regressão ARIMA

Regras do Jogo

- Trabalho 1: 30%
- Prova: 70%

Sumário

Introdução as Séries Temporais e Projeção - Definição

Uma série temporal é uma coleção de observações feitas sequencialmente ao longo do tempo. A característica mais importante deste tipo de dados é que as observações vizinhas são dependentes e estamos interessados em analisar e modelar esta dependência.

Enquanto em modelos de regressão, por exemplo, a ordem das observações é irrelevante para a análise, em séries temporais a ordem dos dados é crucial.

Introdução as Séries Temporais e Projeção - Uso

A projeção de séries temporais é realizada em quase todas as organizações que trabalham com dados quantificáveis:

- Varejistas projetam vendas;
- Companhias de energia projetam reservas, produção, demanda e preços;
- Bancos projetam crescimento de operações de crédito, depósitos, inadimplência, curvas de indicadores financeiros, ...;
- Empresas de transporte usam séries temporais para projetas demandas de viagens futuras.

Introdução as Séries Temporais e Projeção - Exemplo

1- Uma empresa de Marketing precisa realizar a projeção de vendas para o seu orçamento 2016 e para isso está assumindo que as vendas do mês seguinte (t+1) dependem das vendas do mês anterior (t), seguindo a seguinte regra:

$$venda_{t+1} = \alpha + \beta venda_t,$$

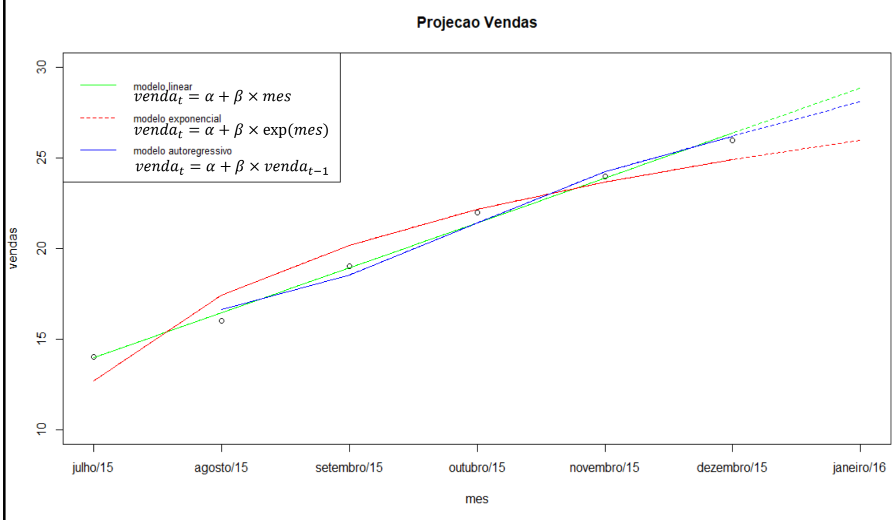
ou seja as vendas do mês t+1 depende das vendas do mês t.

t (mês)	Vendas (milhares)
Julho	14
Agosto	16
Setembro	19
Outubro	22
Novembro	24
Dezembro	26

Calcule uma estimativa para as vendas de Janeiro 2016.

Introdução as Séries Temporais e Projeção - Exemplo

FGV



Introdução as Séries Temporais e Projeção - Notação

FGV

Nesta disciplina iremos manter a notação matemática ao mínimo necessário. Desta forma, a seguinte notação será utilizada ao longo desta disciplina:

Notação	Significado
$t = 1,2,3,\ldots$	Um índice definindo o período de tempo de interesse. $t = 1$ é o primeiro período numa série temporal. Por exemplo, para uma série diária de preços de ações $t = 1,2,3$ definem dia 1, dia 2 e dia 3.
$y_1, y_2, y_3, \ldots, y_n$	Uma série de y valores mensurados durante n períodos , onde y_t define o valor da série temporal no tempo t. Por exemplo, para uma série de preços de ações y_1, y_2, y_3 definem os preços das ações nos dias 1, 2 e 3.
F_t	Valor projetado no período t.
F_{t+k}	Valor projetado k-períodos a frente do momento t. Por exemplo, o valor projetado da ação daqui a 1 dia é representado por F_{t+1}
e_t	Erro de projeção no período t, ou seja, a diferença entre o valor observado e o valor projetado, $e_t = y_t - F_t$.

Introdução as Séries Temporais e Projeção – Processo de Projeção

FGV

O processo de projeção começa com a definição de um objetivo de projeção. Dados são então coletados e explorados utilizando ferramentas de visualização. Um conjunto de métodos de projeção candidatos são selecionados com base na natureza dos dados. Os diferentes métodos são aplicados e comparados em termos de acurácia e consistência com relação ao objetivo de projeção. O melhor método é selecionado e utilizado na projeção final.

Definir objetivo de projeção
Obter Dados
Análise Exploratória dos dados
Pré-processamento dos dados
Particionamento da série temporal
Aplicar métodos de projeção
Avaliar & Comparar performance
Implementar modelo de projeção

Introdução as Séries Temporais e Projeção – Tipos de Análise

FGV

A modelagem de séries temporais pode ser realizada com objetivos descritivos ou projetivos.

Modelagem Descritiva	Modelagem Projetiva
• A série temporal é modelada para determinar os componentes em termos de sazonalidade, tendência, fatores externos; • Utilizada na tomada de decisão e formulação de políticas; • Utiliza modelos para explicar a relação de causa/efeito entre a série temporal e suas variáveis explicativas; • Pode utilizar variáveis passadas, contemporâneas e futuras para explicar o comportamento presente da série. • Exemplos: análises de padrões de compra/venda, fatores que explicam volatilidade de ações...	• Usa informação da série temporal para projetar futuros valores da série; • Utilizada para modelar o melhor comportamento futuro da série temporal com as informações disponíveis até o momento; • Como o foco não é “explicar” a série, modelos caixa-preta são comumente utilizados; • Pode utilizar apenas variáveis passadas e presentes, pois o futuro é o objeto a ser previsto. • Exemplos: projeção de preços, custo, demanda, utilização de equipamentos...

Introdução as Séries Temporais e Projeção – Horizonte de Projeção

1. Quão longe no futuro devemos projetar?
2. Devemos gerar projeções apenas num ponto do tempo ou a projeção deve ser realizada de forma contínua?
3. O quão recentes devem ser dados utilizados para construir a projeção?

- A resposta para as duas perguntas acima dependem diretamente da maneira como as projeções serão utilizadas na prática e necessitam da participação de todos os stakeholders dos resultados da projeção.

- O horizonte de projeção k é o número de períodos futuros que devemos projetar e F_{t+k} é o valor projetado k períodos a frente.

- Por exemplo, assumindo uma série de vendas mensais, a projeção de um mês a frente (F_{t+1}) seria suficiente para questões de precificação de produtos. Caso o objetivo fosse definição do head-count da força de vendas projeções como 3 a 6 meses (F_{t+3} e F_{t+6}) seriam mais apropriados.

- Quanto mais distante no futuro as projeções maior o erro da projeção. Portanto, os modelos devem ser atualizados (treinados) tempestivamente.

- A definição de um limite para o erro do modelo pode ser utilizado como gatilho para a atualização.

Sumário

Dados no contexto de séries temporais

Nesta seção analisaremos:

1. Coleta de dados;
2. Componentes das séries temporais;
3. Visualizando séries temporais;
4. Pré-processamento de dados.

Dados no contexto de séries temporais: 1. Coleta de dados

A decisão de quais dados utilizar para a geração de produções deve levar em consideração vários aspectos, entre eles destacamos:

1. Qualidade dos dados;
2. Frequência temporal;
3. Granularidade;
4. Conhecimento técnico sobre os dados

Dados no contexto de séries temporais – 1.1 Qualidade do dados

- A qualidade dos dados em termos de mensuração da acurácia, valores missing, dados corrompidos e erros de entradas de dados pode afetar substancialmente os resultados da projeções;
- A qualidade dos dados é especialmente importante quando as amostras são pequenas, tipicamente o caso das séries temporais;
- Se existem múltiplas fontes coletando ou armazenando os dados (por exemplo, depto comercial e financeiro armazenam as informações de preços e descontos), é importante comparar os dados de ambos as fontes e escolher a de melhor qualidade como fonte das análises e projeções;
- Normalmente a projeção não é apenas um exercício pontual, precisando ser repetido inúmeras vezes no futuro. Portanto, o processo de governança dos dados é fundamental e contínuo.

Dados no contexto de séries temporais – 1.2 Frequência Temporal

- No atual estado tecnológico muitas séries de dados são registradas em séries temporais de alta frequência. Preços de ações estão disponíveis minuto a minuto. Compras no varejo são registradas em tempo real.
- Entretanto, embora os dados estejam disponíveis em alta frequência, para o propósito de projeção talvez uma frequência menor seja desejável, devido ao nível de ruído nos dados;
- Por exemplo, se o objetivo de projeção são o total de vendas do próximo dia, a melhor informação poderia ser o total de vendas nos dias anteriores e não as vendas minuto a minuto;
- O processo de agregação de dados resulta no cancelamento de ruídos individuais e no ruído geral dos dados, aumentando a precisão das projeções.

Dados no contexto de séries temporais – 1.3 Granularidade

- Granularidade é referente a cobertura dos dados. A cobertura pode estar relacionada a região geográfica, população, tempo da operação, produto, sub-produto, etc...
- Da mesma forma como na definição de frequência, a escolha da granularidade deve estar relacionada com o objetivo de projeção.
- Uma granularidade muito pequena pode levar a "buracos" na série temporal. Por exemplo, um determinado grupo de produto pode vendar de forma frequente, mas um sub-produto específico pode apresentar um padrão concentrado em algumas épocas do ano...

Dados no contexto de séries temporais – 1.4 Conhecimento técnico

- Para conduzir uma análise de séries temporais é fundamental o conhecimento técnico sobre os processos naturais, econômicos e sociais descritos pelos dados;
- O conhecimento técnico é fundamental para determinar a frequência e granularidade dos dados, bem como analisar a performance das projeções.

Dados no contexto de séries temporais – 2. Componentes de séries temporais

- Um série temporal pode ser separada numa parte sistemática e numa não sistemática;
- A parte sistemática é tipicamente dividida em 3 componentes: nível ($Level$), tendência ($Trend$) e sazonalidade ($Seasonality$);
- A parte não sistemática é chamada de ruído ($Noise$);

Componente	Descrição
$Level$	Descreve o valor médio da série temporal.
$Trend$	Descreve a mudança da série entre períodos subsequente.
$Seasonality$	Descreve um comportamento cíclico de curto prazo que pode ser observado repetidamente numa série.
$Noise$	É a variação aleatória que resulta de erros de medição ou outras causa não capturadas nas partes sistemáticas.

Dados no contexto de séries temporais – 2. Componentes de séries temporais

- Os quatro componentes de uma série temporal não observados diretamente. O que observamos é a sua combinação que pode ser de natureza aditiva (onde os componentes variam conforme um valor constante):

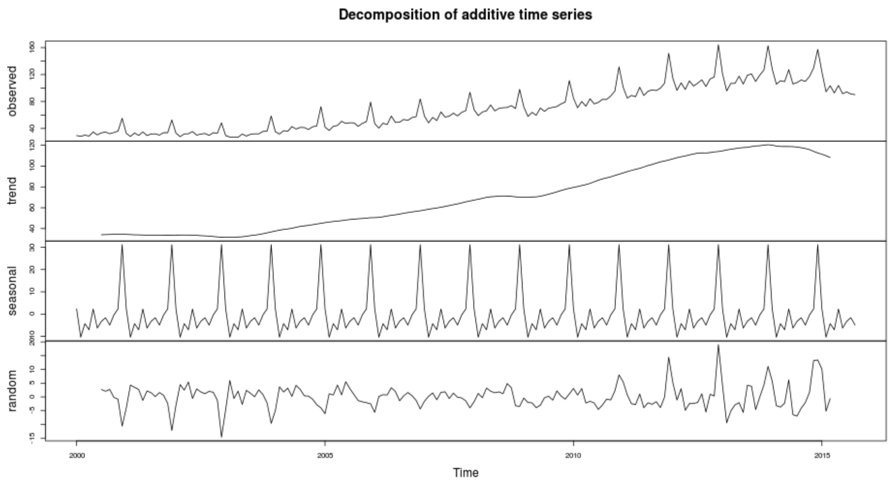
$$y_t = Level + Trend + Seasonality + Noise$$

- Ou multiplicativa (onde os componente variam conforme um percentual):

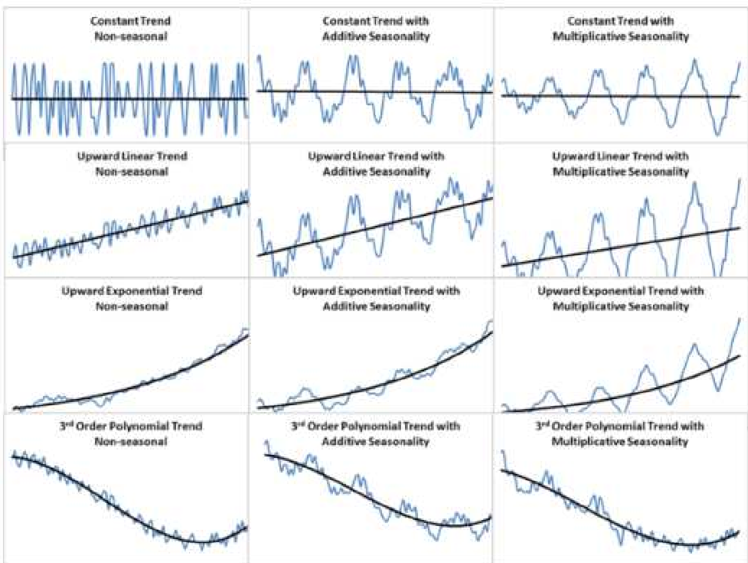
$$y_t = Level \times Trend \times Seasonality \times Noise$$

- Métodos de projeção buscam isolar a parte sistemática e quantificar o nível de ruído (efeito não sistemático na projeção).

Dados no contexto de séries temporais – 2. Componentes de séries temporais



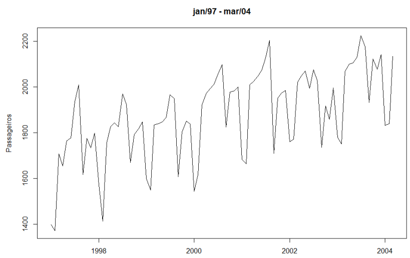
Dados no contexto de séries temporais – 2. Componentes de séries temporais



Dados no contexto de séries temporais – 3. Visualizando séries temporais

- Um efetivo primeiro passo para caracterizar a natureza de uma série temporal e detectar potenciais erros é fazer uso da visualização de dados.
- Visualizando os dados conseguimos identificar padrões iniciais, identificar os seus componentes e verificar potenciais problemas como valores extremos, espaçamento desigual e valores missing.
- A forma visual mais básica e informativa é o gráfico temporal da série.

Month	Ridership
jan/91	1709
fev/91	1621
mar/91	1973
abr/91	1812
mai/91	1975
jun/91	1862
jul/91	1940
ago/91	2013
set/91	1596
out/91	1725
nov/91	1676
dez/91	1814
jan/92	1615
fev/92	1557
mar/92	1891



Dados no contexto de séries temporais – 4. Pré-processamento de dados

Após a inspeção visual da série de dados, os problemas detectados podem ser ajustados numa etapa chamada de pré-processamento. Alguns dos principais problemas e soluções são:

- **Valores Missing:** valores faltantes criam buracos na série temporal e podem afetar os modelos de projeção de forma variada. Uma solução simples para este problema é preencher os valores missing com alguma estimativa. Por exemplo, a média entre os dois pontos mais próximos. Outros exemplos constituem na criação de um modelo para estimar valores faltantes ou utilizar dados externos para completar a base;

- **Series com espaçamentos diferentes:** séries com espaçamentos iguais são aquelas onde o valores são capitados e ocorrem na mesma frequência, por exemplo, os preços de ações são listados a cada 1 minuto. Porém, os pedidos de compras de ações não precisam ocorrer a cada minuto. Neste caso uma série minuto a minuto apresentaria buracos, onde compras não ocorreram. Soluções similares a dos valores missing podem ser aplicadas ou a consolidação em frequências menores como 15 min ou 1 hora...

Dados no contexto de séries temporais – 4. Pré-processamento de dados

Após a inspeção visual da série de dados, os problemas detectados podem ser ajustados numa etapa chamada de pré-processamento. Alguns dos principais problemas e soluções são:

- **Valores extremos:** são valores muito grandes ou pequenos quando comparados as demais valores da série. Valores extremos podem afetar os modelos de projeção de forma variadas. A decisão de remover valores extremos deve estar embasada no conhecimento do comportamento natural, econômico ou social que a série representa. O valor pode ser resultado de um erro operacional de mensuração? O valor representa um momento atípico que não irá mais ocorrer na série de dados? Caso não existe uma justificativa plausível para remover o valor, uma abordagem interessante é modelar a série com os valore e sem os valores extremos e compara os resultados.

Sumário

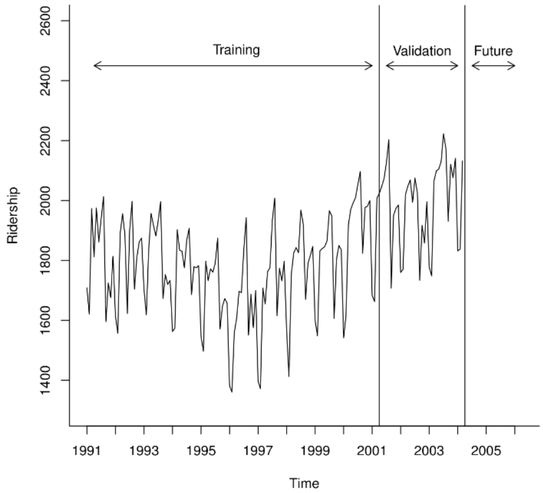
Avaliação de Performance

- Numa primeira abordagem, a melhor abordagem seria escolher o modelo que gera a melhor projeção considerando toda a série temporal disponível;
- Entretanto, quando utilizamos os mesmos dados para desenvolver o modelo de projeção e para avaliar a sua performance acabamos por introduzir um viés;
- Quando escolhemos um modelo com base na maior aderência a toda a nossa amostra estamos observado dois aspectos:
  1. Um modelo com aderência superior;
  2. Coincidência de que o modelo "casa" com os dados observados.
- O segundo aspecto é um problema grave para modelos de projeção que não impõem uma estrutura nos dados (por exemplo, os dados de vendas são lineares ou exponencial), pois eles acabam resultando em overfitting dos dados;
- Overfitting significa que não estamos apenas modelando a parte sistemática dos dados (nível, tendência e sazonalidade), mas também o ruído (noise).
- Este aspecto pode resultar num modelo com baixa acurácia em dados fora da amostra. Exatamente o futuro que queremos projetar.

Avaliação de Performance – Particionamento dos Dados

- Para evitar o overfitting e permitir a análise da performance do modelo em novos dados é necessário a separação da série de dados em dois períodos: treinamento e validação.
- As duas partições são criadas mantendo a sequencia temporal da série de dados onde:
  - Período de Treinamento: $y_1, y_2, y_3, \cdots, y_n$
  - Período de Validação: $y_{n+1}, y_{n+2}, y_{n+3}, \cdots, y_{n+k}$
- Os modelos são estimados utilizando o período de treinamento e analisados sobre ótica de performance no período de validação.

Avaliação de Performance – Particionamento dos Dados



Avaliação de Performance – Combinando partições de dados

Após a seleção do melhor modelo para projeção, os dados de treinamento e validação são recombinados e o modelo escolhido estimado com os dados totais. Este método possui três vantagens:

1. O período de validação, normalmente o mais recente possui a informação mais relevante para a projeção do futuro, pois está temporalmente mais próximo;

2. Com mais dados os modelos podem ser estimados com maior precisão;

3. Se apenas o período de treinamento for utilizado para gerar projeções do futuro estaremos aumentado o horizonte de projeções. Por exemplo, se o período de validação possui 4 pontos de dados, projetar a primeira observação do futuro a partir dos dados de treinamento significa fazer uma projeção de 5 períodos a frente, ou seja, F_{t+5}. Aumentando consideravelmente a incerteza.

Avaliação de Performance – Escolhendo o Período de Validação

- Escolher o tamanho do período de validação depende do objetivo de projeção, da frequência dos dados e do horizonte de projeção;
- A ideia principal é escolher um período de validação que seja o mais próximo do comportamento esperado para o período de projeção, com objetivo de utilizar um benchmark correto para a performance do modelo;
- Por exemplo, se o objetivo for projetar um ano de vendas de determinado produto o período de validação deveria conter no mínimo 1 ano.
    - Períodos de validação mais curtos não iram capturar os efeitos de sazonalidade e tendência anual.
    - Períodos mais longos resultariam em projeções muito distantes para as quais o período de treinamento não possui pouca informação.

Avaliação de Performance – Benchmark de projeção - Naive

Um bom benchmark para comparar modelos de projeção é o chamado de projeção Naive, algo como projeção ingênua. Trata-se de um modelo simples e que considera como melhor estimador para todos os valores futuros a informação mais recente. Ou seja, para k períodos a frente o modelo Naive é:

$$F_{t+k} = y_t$$

Onde y_t é o último valor observado da série.

A utilização de modelos Naive serve a dois propósitos:

1. Modelos Naive são simples de implementar e, algumas vezes obtêm resultados adequados de performance. Considerando o trade-off entre simplicidade e acurácia o modelo Naive é um bom candidato.

2. O modelo Naive funciona como benchmark para modelos mais sofisticados. Indicando se a complexidade do modelo entrega resultados melhores que um estimador simples.

Avaliação de Performance – Mensurando Acurácia de Projeção

- Como o objetivo de projeção é a geração de projeções precisas, precisamos definir alguma métrica para avaliar a acurácia;
- Tais métricas informam como um método de projeção performar em particular, bem como quando comparado com benchmarks ou projeções de outros modelos;
- É necessário enfatizar que quando tratamos de acurácia de precisão não estamos preocupados com a aderência aos dados de treinamento (o popular R^2), mas sim com os erros de **<u>projeção no período de validação</u>**:

$$e_t = y_t - F_t$$

Onde $t \in \{n, n+1, n+2, \dots, n+k\}$ e k é a quantidade total de dados na amostra de validação.

Avaliação de Performance – Mensurando Acurácia de Projeção

Medida	Fórmula	Significado
Erro Médio	$\frac{1}{k}\sum_{t=1}^{k} e_t$	Calcula a média dos erros cancelando os erros positivos e negativos. Pode ignorar a magnitude dos desvios dos erros.
Erro Médio Quadrado (MSE)	$\frac{1}{k}\sum_{t=1}^{k} e_t^2$	Calcula o erro médio quadrado. Ao contrário do erro médio ele considera a distância de cada erro, amplificando os erros maiores.
Raiz do Erro Médio Quadrado (RMSE)	$\sqrt{\frac{1}{k}\sum_{t=1}^{k} e_t^2}$	Igual ao erro médio, mas padronizando a medida para a mesma escala da série temporal.
Erro Médio Absoluto (MAE)	$\frac{1}{k}\sum_{t=1}^{k} \lvert e_t \rvert$	Calcula o erro médio absoluto com características similares ao RMSE
Erro Médio Absoluto Percentual (MAPE)	$\frac{1}{k}\sum_{t=1}^{k} \left\lvert \frac{e_t}{y_t} \right\rvert \times 100$	Esta medida calcula um score percentual de como as projeções variam em média dos valores atuais. Métrica útil para compara a performance entre séries que apresentam diferentes escalas.

Avaliação de Performance – Acurácia de Projeção X Lucratividade

- Em alguns contextos a medida de acurácia da projeção não é adequada de maneira individual e deve ser combinada com os custos do erro para gerar uma medida de lucratividade;
- O custo dos erros de projeção podem ser diferentes para erros positivos (projeção acima do valor observado) e negativos (projeção abaixo do valor observado), ou mais sensíveis numa direção do que outra;
- Por exemplo, erro de projeção no preço de uma ação pode levar o investidor a comprar a ação e perder principal, ou não comprar a ação e perder o ganho de rentabilidade da oportunidade;
- Outro exemplo, erro na projeção de demanda pode levar a um excesso de produção, aumentando os custos de estoque ou a indisponibilidade do produto para havidos consumidores, representando uma perda de receita potencial.

Sumário

Métodos de Projeção: Visão Geral

1. **Métodos Baseados em modelos (model based):**
   - ✓ Utilizam modelos estatísticos, matemáticos ou outro modelo científico que aproxima os dados da série;
   - ✓ Os dados de treinamento são utilizados para estimar os parâmetros do modelo e então com os parâmetros estimados a projeção é realizada;
   - ✓ Exemplos de modelos são: regressões lineares e modelos autoregressivos onde o usuário especifica um modelo linear ou não-linear para estimar a série temporal;
   - ✓ Modelos baseados em métodos são úteis quando a série temporal é curta, pois partindo da premissa de um modelo de comportamento (por exemplo, linear) precisamos de poucos dados para estimar o modelo;
   - ✓ Método normalmente utilizado para identificar padrões globais;

2. **Métodos Baseados em dados (data based):**
   - ✓ Modelos onde o algoritmo de estimação aprende padrões dos dados, sem forçar uma estrutura de comportamento dados (por exemplo, linear);
   - ✓ Trata-se de um método recomendado quando as premissas do modelo provavelmente serão violadas ou quando a estrutura da série temporal muda ao longo do tempo;
   - ✓ Modelos de Suavização e Naive são orientados a dados;
   - ✓ São modelos que demandam poucas definições por parte do usuário, pois aprendem com os dados;
   - ✓ Como são modelos que aprendem com os dados necessitam de séries temporais mais longas;
   - ✓ Modelos aprendidos dos dados podem ser extremamente complexos (caixas pretas);
   - ✓ Método normalmente utilizado para identificar padrões locais;

Sumário

- Apresentações
- Regras do Jogo
- Introdução as Séries Temporais e Projeção
- Dados no contexto de Séries Temporais
- Avaliação de Performance
- Métodos de Projeção: Visão Geral
- **Métodos de Regressão**
- Métodos de Projeção: Suavização
- Modelos de Regressão ARIMA

Modelos de Regressão – Modelos com Tendência

- Um modelo popular de projeção é baseado em modelos de regressão linear, usando variáveis explicativas para capturar tendência e sazonalidade, bem como outros padrões dos dados;
- O modelo de regressão linear pode ser utiliza para estimar uma tendência global que é aplicável para toda a série temporal e será aplica no período de projeção;
- Uma tendência linear indica que os valores da série temporal aumentam ou diminuem de forma linear ao longo do tempo, para tendência exponencias as mudanças ocorrem da mesma maneira, porém exponencialmente;
- Além de tendências lineares e exponenciais podemos utilizar outros modelos como: quadrático, cúbico, logarítmico, etc...

Modelos de Regressão – Modelos com Tendência: 1. Linear

- Para criar um modelo de regressão linear que captura uma série temporal com tendência linear global, a variável dependente y é alinhada com uma variável explicativa x definida como um índice temporal.
- Por exemplo, a variável dependente y será a quantidade de passageiros (ridership) e variável explicativa x um contador t para indicar cada período.

$$y_t = \beta_0 + \beta_1 t + e_t$$

- Com a equação acima podemos estimar 3 componentes da série temporal: nível (β_0), tendência (β_1) e ruído (e_t), apenas a sazonalidade não é modelada.

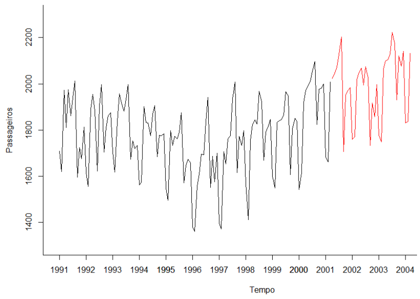
Month	Ridership	t
Jan-91	1708.917	1
Feb-91	1620.586	2
Mar-91	1972.715	3
Apr-91	1811.665	4
May-91	1974.964	5
Jun-91	1862.356	6
Jul-91	1939.860	7
Aug-91	2013.264	8
Sep-91	1595.657	9
Oct-91	1724.924	10
Nov-91	1675.667	11
Dec-91	1813.863	12

Modelos de Regressão – Modelos com Tendência: 1. Linear

FGV

- Definido o modelo precisamos separar a série temporal em amostra de treinamento e validação.

```
#define o tamanho da amostra de teste
tam_amostra_teste <- 36

#define o tamanho da amostra de treinamento
tam_amostra_treinamento <- length(passageiros_ts) - tam_amostra_teste

#cria a serie temporal de treinamento
treinamento_ts <- window(passageiros_ts, start=c(1991, 1), end=c(1991,tam_amostra_treinamento))

#cria a serie temporal de teste
treinamento_ts <- window(passageiros_ts, start=c(1991, tam_amostra_treinamento + 1), end=c(1991,tam_amostra_treinamento+tam_amostra_teste))
```

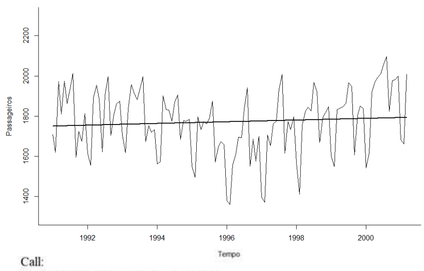


Modelos de Regressão – Modelos com Tendência: 1. Linear

FGV

- Depois calibramos o modelo de tendência linear na série temporal de treinamento

```
#Estima o modelo de tendência linear
modelo_tendencia_linear <- tslm(treinamento_ts ~ trend)

#resumo do modelo
summary(modelo_tendencia_linear)

#plot modelo com tendencia
plot(treinamento_ts, xlab="Tempo", ylab="Passageiros", ylim=c(1300, 2300), bty="l")
lines(modelo_tendencia_linear$fitted.values, lwd=2)
```

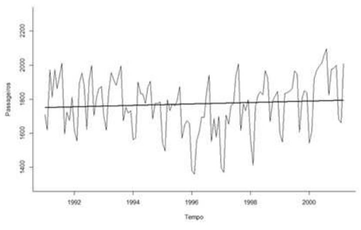


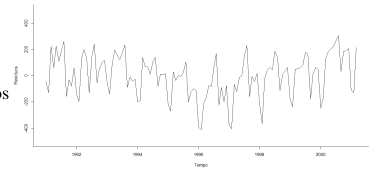

```
Call:
tslm(formula = treinamento_ts ~ trend)

Residuals:
    Min      1Q  Median      3Q     Max
-411.29 -114.02   16.06  129.28  306.35

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 1750.3595    29.0729  60.206   <2e-16 ***
trend          0.3514     0.4069   0.864     0.39
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 160.2 on 121 degrees of freedom
Multiple R-squared:  0.006125,	Adjusted R-squared:  -0.002089
F-statistic: 0.7456 on 1 and 121 DF,  p-value: 0.3896
```

Modelos de Regressão – Modelos com Tendência: Análise de Resíduos (Noise)

- Os resíduos (ruído ou noise) numa série temporal são os valores remanescentes após a o fit de um modelo, ou seja, são iguais as diferenças entre os valores observados e os valores estimados:
  $$e_t = y_t - \hat{y}_t$$
- Resíduos são úteis para checar se um modelo capturou adequadamente a informação nos dados. Um bom modelo de projeção resultará em resíduos com as seguintes características:

  1. Os resíduos não são correlacionados. Se existirem correlações entre os resíduos então existe informação útil no resíduo que deveria ser utilizada na projeção;
  2. Os resíduos possuem média zero. Se os resíduos não possuem média zero então as projeções são enviesadas.





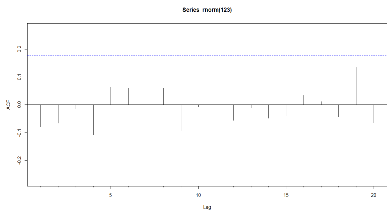
- Qualquer método de projeção que não satisfaça estas características pode ser melhorado. Isto não significa que métodos que atendem estas características não podem ser melhorados.
- Adicionalmente aos dois aspectos acima, é útil para estimação dos intervalos de confiança, mas não necessário, que os resíduos apresentem as seguintes características:

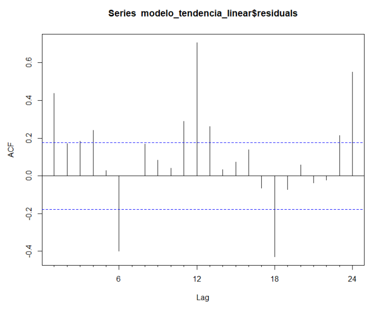
  3. Resíduos possuem variância constante;
  4. Os resíduos sejam normalmente distribuídos.

Modelos de Regressão – Modelos com Tendência: Análise de Resíduos (Noise) - ACF

- Primeiramente analisamos se os resíduos são correlacionados.
- Para isso utilizamos uma função de autocorrelação (ACF) que dentro de um intervalo de confiança ($\mp \frac{2}{\sqrt{T}}$ para 95% de intervalo de confiança) identifica se os valores da séries são correlacionados de forma estatisticamente significante.

```
#Verificando resíduos
Acf(modelo_tendencia_linear$residuals)
```





- Conforme a figura ao lado a série apresenta significativo grau de correlação positiva, em grande parte devido aos efeitos de tendência e sazonalidade.

$$r_k = \frac{\sum_{t=1}^{n-k}(x_t - \overline{x})(x_{t+k} - \overline{x})}{\sum_{t=1}^{n}(x_t - \overline{x})^2}$$

Modelos de Regressão – Modelos com Tendência: Análise de Resíduos (Noise) – Ljung-Box Test

- Em adição as análises de ACF, nós podemos realizar um teste mais formal para autocorrelação considerando todos os para de correlação $r_k$estimados ao invés de cada valor de forma individual como no ACF.
- Quando analisamos um gráfico ACF procuramos garantir que nenhum de r_k extrapola os limites para o intervalo de confiança definido. Esta análise é realizada para todos os lags em análise e pode levar a falsos positivos, podendo resultar numa falsa conclusão de autocorrelação.
- Para resolver este problema, podemos realizar um teste conjunto se as primeiras k correlações são estatisticamente diferentes de um processo de ruído branco. Um teste utilizado para este propósito é o Ljung-Box test:

$$Q = T(T+2)\sum_{k=1}^{h}(T-k)^{-1}r_k^2$$

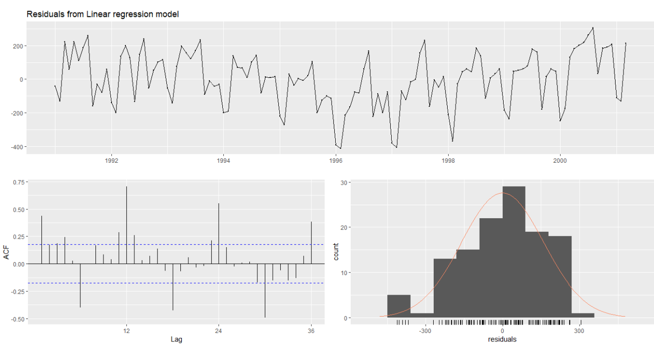
- Onde h é o valor de lags e k o número de parâmetros do modelo.
- Um valor grande para Q significa que os valores de autocorrelação não são oriundos de um processo de ruído branco. Como Q segue uma distribuição de Chi-Square (χ^2). Podemos utilizar um teste de p-value para definir o resultado. Lembrando que quanto menor o valor do p-value confirma-se a hipótese de autocorrelação.

Modelos de Regressão – Modelos com Tendência: Análise de Resíduos (Noise) – checkresiduals

- O R possui uma função chamada checkresiduals() que computa os valores de ACF e testes estatísticos de forma automática.

```
#verifica os resíduos com teste de Ljung-Box
checkresiduals(modelo_tendencia_linear, test="LB")
```

```
Ljung-Box test

data:  Residuals from Linear regression model
Q* = 242.28, df = 22, p-value < 2.2e-16

Model df: 2.   Total lags used: 24
```

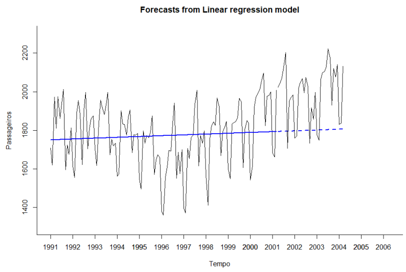


Modelos de Regressão – Modelos com Tendência: 1. Linear

FGV

- Depois testamos o modelo na série temporal de validação

```
#projeta o modelo durante o período de validação
modelo_tendencia_linear_proj <-
forecast(modelo_tendencia_linear, h =
tam_amostra_teste, level=0)

#plota o grafico da serie temporal de treinamento e teste
plot(modelo_tendencia_linear_proj, xlab="Tempo",
ylab="Passageiros", xant="n" , ylim=c(1300, 2300),
xlim=c(1991, 2006.25), bty="l", flty=2)

axis(1, at=seq(1991, 2006, 1), labels=format(seq(1991,
2006,1)))

lines(validacao_ts)
lines(modelo_tendencia_linear_proj$fitted, lwd=2,
col="blue")

#Verifica a acuracia do modelo
accuracy(modelo_tendencia_linear_proj, validacao_ts)
```



Modelo de Tendência Linear

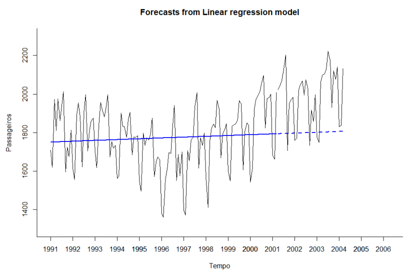
	ME	RMSE	MAE	MPE	MAPE
Training set	- 0,0000	158,9269	129,6778	- 0,8540	7,5360
Test set	193,1316	239,4863	209,4371	9,2099	10,1477

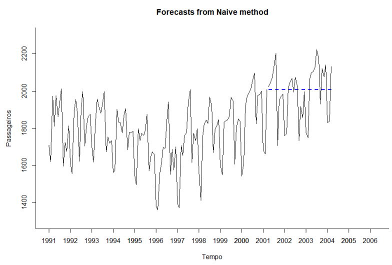
Modelos de Regressão – Modelos com Tendência: 1. Linear

FGV

- Comparando com o Modelo Naive

```
#Calula o modelo naive
modelo_naive <- naive(treinamento_ts, level=0,
h=tam_amostra_teste)

#Verifica a acuracia do modelo naive
accuracy(modelo_naive, validacao_ts)

#plota o grafico da serie temporal de treinamento e teste
plot(modelo_naive, xlab="Tempo", ylab="Passageiros",
xant="n" , ylim=c(1300, 2300), xlim=c(1991, 2006.25),
bty="l", flty=2)

axis(1, at=seq(1991, 2006, 1), labels=format(seq(1991,
2006,1)))

lines(validacao_ts)
```





Modelo de Tendência Linear

	ME	RMSE	MAE	MPE	MAPE
Training set	- 0,0000	158,9269	129,6778	- 0,8540	7,5360
Test set	193,1316	239,4863	209,4371	9,2099	10,1477

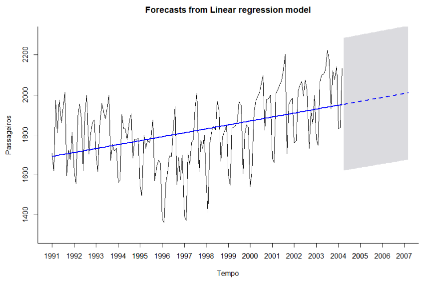
Modelo de Tendência Naive

	ME	RMSE	MAE	MPE	MAPE
Training set	2,4509	168,1470	125,2975	- 0,3460	7,2714
Test set	- 14,7177	142,7551	115,9234	- 1,2750	6,0214

Modelos de Regressão – Modelos com Tendência: 1. Linear

- Projetar para os próximos 36 meses no futuro

```
#projetar para os próximos 36 meses no futuro

#primeiramente reestimamos o modelo com todos os
dados de treinamento e validacao
modelo_tendencia_linear_final <- tslm(passageiros_ts ~
trend)

#sumario do modelo
summary(modelo_tendencia_linear_final)

#projeta os próximos 36 meses do futuro
modelo_tendencia_linear_final_proj <-
forecast(modelo_tendencia_linear_final, h=36,
level=0.95)

#plota o grafico da serie temporal de treinamento e teste
plot(modelo_tendencia_linear_final_proj, xlab="Tempo",
ylab="Passageiros", ylim=c(1300, 2300), xlim=c(1991,
2007), bty="l", flty=2)
axis(1, at=seq(1991, 2007, 1), labels=format(seq(1991,
2007,1)))
lines(modelo_tendencia_linear_final_proj$fitted, lwd=2,
col="blue")
```



Modelos de Regressão – Modelos com Tendência: 2. Polinomial

- Para criar um modelo de regressão polinomial que captura uma série temporal com tendência linear global, a variável dependente y é alinhada com uma variável explicativa x definida como um índice temporal.

- Por exemplo, a variável dependente y será a quantidade de passageiros (ridership) e variável explicativa x um contador t para indicar cada período, no exemplo quadrático termos:

$$y_t = \beta_0 + \beta_1 t + \beta_2 t^2 + e_t$$

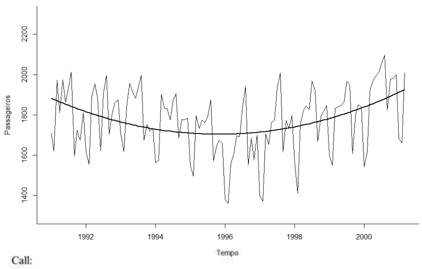
- Com a equação acima podemos estimar 3 componentes da série temporal: nível (β_0), tendência (β_1 e β_2) e ruído (e_t), apenas a sazonalidade não é modelada.

Month	Ridership	t	t2
Jan-91	1708.917	1	1
Feb-91	1620.586	2	4
Mar-91	1972.715	3	9
Apr-91	1811.665	4	16
May-91	1974.964	5	25
Jun-91	1862.356	6	36
Jul-91	1939.860	7	49
Aug-91	2013.264	8	64
Sep-91	1595.657	9	81
Oct-91	1724.924	10	100
Nov-91	1675.667	11	121
Dec-91	1813.863	12	144

Modelos de Regressão – Modelos com Tendência: 2. Polinomial

- Depois calibramos o modelo de tendência linear na série temporal de treinamento

```
#Estima o modelo de tendência poli
modelo_tendencia_poli <- tslm(treinamento_ts ~ trend + I(trend^2))

#resumo do modelo
summary(modelo_tendencia_poli)

#plot modelo com tendencia
plot(treinamento_ts, xlab="Tempo", ylab="Passageiros", ylim=c(1300, 2300), bty="l")
lines(modelo_tendencia_poli$fitted.values, lwd=2)
```



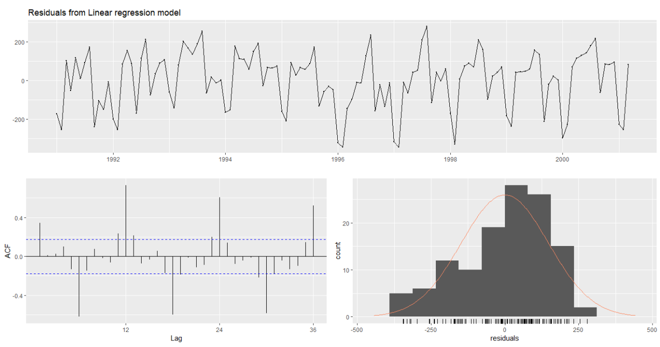

```
Call:
tslm(formula = treinamento_ts ~ trend + I(trend^2))

Residuals:
    Min      1Q  Median      3Q     Max 
-344.79 -101.86   40.89   98.54  279.81 

Coefficients:
              Estimate Std. Error t value Pr(>|t|)    
(Intercept) 1888.88401   40.91521  46.166  < 2e-16 ***
trend         -6.29780    1.52327  -4.134 6.63e-05 ***
I(trend^2)     0.05362    0.01190   4.506 1.55e-05 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 148.8 on 120 degrees of freedom
Multiple R-squared:  0.1499,	Adjusted R-squared:  0.1358 
F-statistic: 10.58 on 2 and 120 DF,  p-value: 5.844e-05
```

Modelos de Regressão – Modelos com Tendência: Análise de Resíduos (Noise) – checkresiduals

- O R possui uma função chamada checkresiduals() que computa os valores de ACF e testes estatísticos de forma automática.

```
#verifica os resíduos com teste de Ljung-Box
checkresiduals(modelo_tendencia_poli, test="LB")
```

```
Ljung-Box test

data:  Residuals from Linear regression model
Q* = 291.69, df = 21, p-value < 2.2e-16

Model df: 3.   Total lags used: 24
```

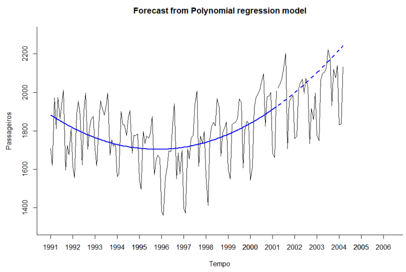


Modelos de Regressão – Modelos com Tendência: 2. Polinomial

FGV

- Depois testamos o modelo na série temporal de validação

```
#projeta o modelo durante o período de validação
modelo_tendencia_poli_proj <-
forecast(modelo_tendencia_poli, h = tam_amostra_teste,
level=0)

#plota o grafico da serie temporal de treinamento e teste
plot(modelo_tendencia_poli_proj, xlab="Tempo",
ylab="Passageiros", xant="n" , ylim=c(1300, 2300),
xlim=c(1991, 2006.25), bty="l", flty=2)

axis(1, at=seq(1991, 2006, 1), labels=format(seq(1991,
2006,1)))

lines(validacao_ts)
lines(modelo_tendencia_poli_proj$fitted, lwd=2,
col="blue")


#Verifica a acuracia do modelo
accuracy(modelo_tendencia_poli_proj, validacao_ts)
```

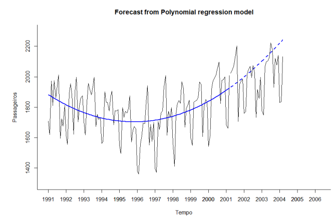


Modelo de Tendência Quadrático					
	ME	RMSE	MAE	MPE	MAPE
Training set	0,0000	146,9785	120,3980	- 0,7350	7,0158
Test set	- 83,9621	179,8494	133,7383	- 4,7254	7,0757

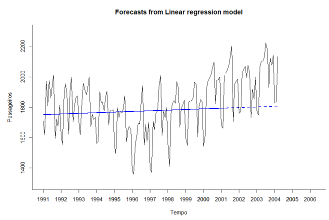
Modelos de Regressão – Modelos com Tendência: 2. Polinomial

FGV

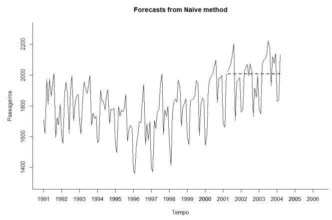
- Comparando com o Modelo Naive e Linear



Modelo de Tendência Quadrático					
	ME	RMSE	MAE	MPE	MAPE
Training set	0,0000	146,9785	120,3980	- 0,7350	7,0158
Test set	- 83,9621	179,8494	133,7383	- 4,7254	7,0757



Modelo de Tendência Linear					
	ME	RMSE	MAE	MPE	MAPE
Training set	- 0,0000	158,9269	129,6778	- 0,8540	7,5360
Test set	193,1316	239,4863	209,4371	9,2099	10,1477

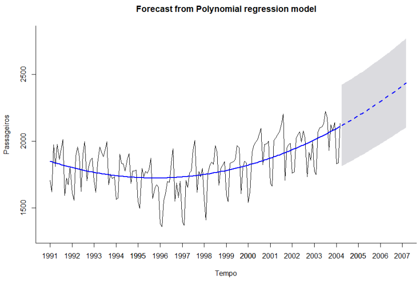


Modelo de Tendência Naive					
	ME	RMSE	MAE	MPE	MAPE
Training set	2,4509	168,1470	125,2975	- 0,3460	7,2714
Test set	- 14,7177	142,7551	115,9234	- 1,2750	6,0214

Modelos de Regressão – Modelos com Tendência: 2. Polinomial

- Projetar para os próximos 36 meses no futuro

```
#projetar para os próximos 36 meses no futuro

#primeiramente reestimamos o modelo com todos os dados de treinamento e validacao
modelo_tendencia_poli_final <- tslm(passageiros_ts ~ trend + I(trend^2))

#sumario do modelo
summary(modelo_tendencia_poli_final)

#projeta os próximos 36 meses do futuro
modelo_tendencia_poli_final_proj <- forecast(modelo_tendencia_poli_final, h=36, level=0.95)

#plota o grafico da serie temporal de treinamento e teste
plot(modelo_tendencia_poli_final_proj, xlab="Tempo", ylab="Passageiros", ylim=c(1300, 2800), xlim=c(1991, 2007), bty="l", flty=2)
axis(1, at=seq(1991, 2007, 1), labels=format(seq(1991, 2007,1)))
lines(modelo_tendencia_poli_final_proj$fitted, lwd=2, col="blue")
```



Modelos de Regressão – Modelos com Tendência: 3. Exponencial

- Para criar um modelo de regressão exponencial que captura uma série temporal com tendência linear global, a variável dependente y é alinhada com uma variável explicativa x definida como um índice temporal.

- Por exemplo, a variável dependente y será a quantidade de passageiros (ridership) e variável explicativa x um contador t para indicar cada período.

$$y_t = ce^{\beta_1 t + e_t}$$

- O formato acima é computacionalmente intensivo, então utilizamos o formato transformado para:

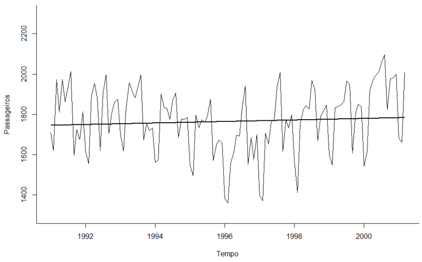
$$\ln y_t = \beta_0 + \beta_1 t + e_t$$

- Com a equação acima podemos estimar 3 componentes da série temporal: nível (β_0), tendência (β_1) e ruído (e_t), apenas a sazonalidade não é modelada.

Month	Ridership	t	t2	ln_Ridership
Jan-91	1708.917	1	1	7.443615
Feb-91	1620.586	2	4	7.390543
Mar-91	1972.715	3	9	7.587166
Apr-91	1811.665	4	16	7.502002
May-91	1974.964	5	25	7.588305
Jun-91	1862.356	6	36	7.529598
Jul-91	1939.860	7	49	7.570371
Aug-91	2013.264	8	64	7.607513
Sep-91	1595.657	9	81	7.375041
Oct-91	1724.924	10	100	7.452938
Nov-91	1675.667	11	121	7.423967
Dec-91	1813.863	12	144	7.503214

Modelos de Regressão – Modelos com Tendência: 3. Exponencial

- Depois calibramos o modelo de tendência exponencial na série temporal de treinamento.
- O R faz a conversão logarítmica de forma automática pelo parâmetro lambda=0.
- Os resultados não são muito diferentes do modelo linear, pois a tendência é suave.




```
#Estima o modelo de tendência exp
modelo_tendencia_exp <- tslm(treinamento_ts ~ trend,
lambda=0)

#resumo do modelo
summary(modelo_tendencia_exp)

#plot modelo com tendencia
plot(treinamento_ts, xlab="Tempo", ylab="Passageiros",
ylim=c(1300, 2300), bty="l")
lines(modelo_tendencia_exp$fitted.values, lwd=2)
```

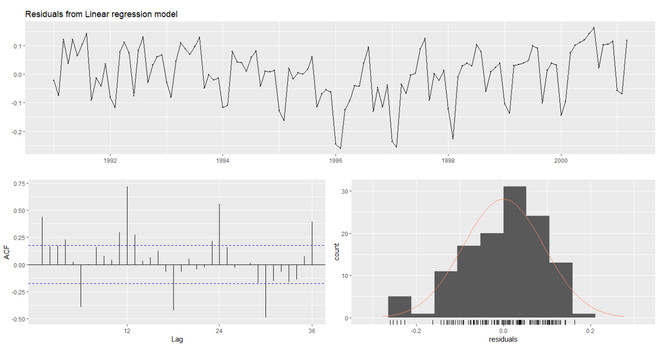
```
Call:
tslm(formula = treinamento_ts ~ trend, lambda = 0)

Residuals:
     Min       1Q   Median       3Q      Max
-0.25989 -0.06184  0.01299  0.07457  0.16310

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 7.4646979  0.0168680 442.535   <2e-16 ***
trend       0.0001783  0.0002361   0.755    0.451
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.09297 on 121 degrees of freedom
Multiple R-squared:  0.9999,	Adjusted R-squared:  0.9999
F-statistic: 1.777e+06 on 1 and 121 DF,  p-value: < 2.2e-16
```

Modelos de Regressão – Modelos com Tendência: Análise de Resíduos (Noise) – checkresiduals

- O R possui uma função chamada checkresiduals() que computa os valores de ACF e testes estatísticos de forma automática.

```
#verifica os resíduos com teste de Ljung-Box
checkresiduals(modelo_tendencia_exp, test="LB")
```

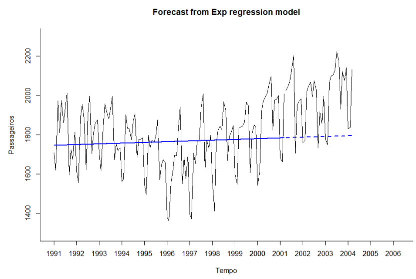
```
Ljung-Box test

data:  Residuals from Linear regression model
Q* = 245.28, df = 22, p-value < 2.2e-16

Model df: 2.   Total lags used: 24
```



Modelos de Regressão – Modelos com Tendência: 3. Exponencial

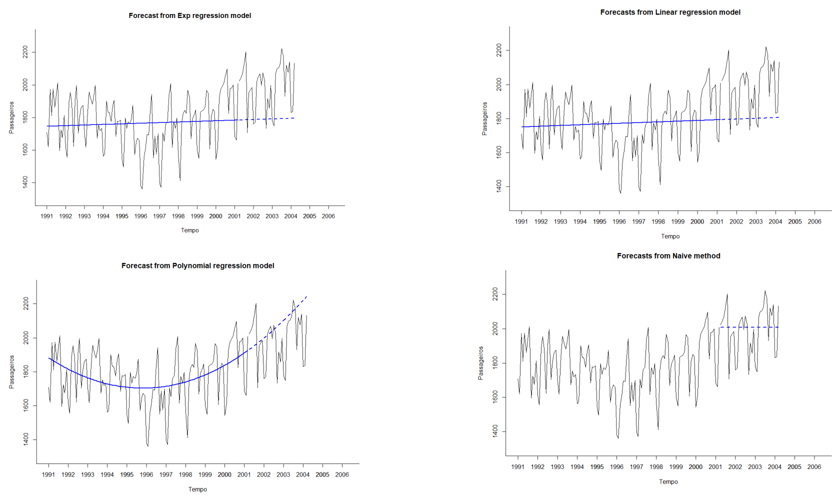
- Depois testamos o modelo na série temporal de validação

```
#projeta o modelo durante o período de validação
modelo_tendencia_exp_proj <-
forecast(modelo_tendencia_exp, h = tam_amostra_teste,
level=0)

#plota o grafico da serie temporal de treinamento e teste
plot(modelo_tendencia_exp_proj, xlab="Tempo",
ylab="Passageiros", xant="n" , ylim=c(1300, 2300),
xlim=c(1991, 2006.25), bty="l", flty=2 ,main="Forecast
from Exp regression model")

axis(1, at=seq(1991, 2006, 1), labels=format(seq(1991,
2006,1)))

lines(validacao_ts)
lines(modelo_tendencia_exp_proj$fitted, lwd=2,
col="blue")


#Verifica a acuracia do modelo
accuracy(modelo_tendencia_exp_proj, validacao_ts)
```



Modelos de Regressão – Modelos com Tendência: 3. Exponencial

- Comparando com o Modelo Naive, Quadrático e Linear



Modelos de Regressão – Modelos com Tendência: 3. Exponencial

- Comparando com o Modelo Naive, Quadrático e Linear

Modelo de Tendência Exponencial					
	ME	RMSE	MAE	MPE	MAPE
Training set	7,3750	159,0912	130,5631	- 0,4346	7,5540
Test set	203,2733	247,7550	216,2353	9,7213	10,4680

Modelo de Tendência Quadrático					
	ME	RMSE	MAE	MPE	MAPE
Training set	0,0000	146,9785	120,3980	- 0,7350	7,0158
Test set	- 83,9621	179,8494	133,7383	- 4,7254	7,0757

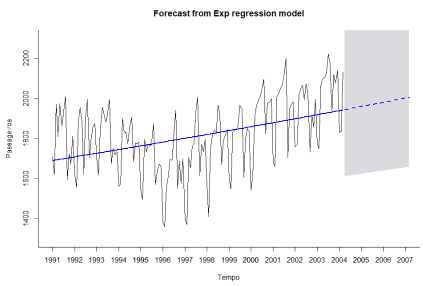
Modelo de Tendência Linear					
	ME	RMSE	MAE	MPE	MAPE
Training set	- 0,0000	158,9269	129,6778	- 0,8540	7,5360
Test set	193,1316	239,4863	209,4371	9,2099	10,1477

Modelo de Tendência Naive					
	ME	RMSE	MAE	MPE	MAPE
Training set	2,4509	168,1470	125,2975	- 0,3460	7,2714
Test set	- 14,7177	142,7551	115,9234	- 1,2750	6,0214

Modelos de Regressão – Modelos com Tendência: 3. Exponencial

- Projetar para os próximos 36 meses no futuro

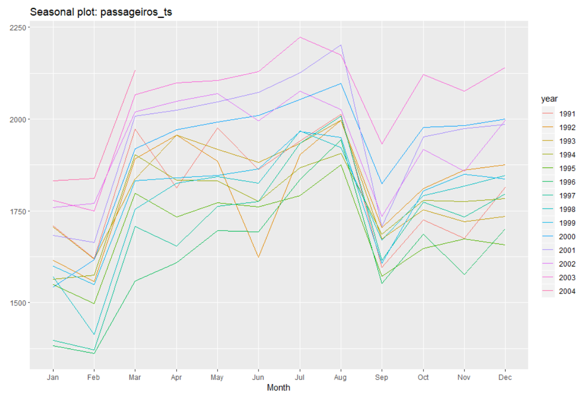
```
#primeiramente reestimamos o modelo com todos os
dados de treinamento e validacao
modelo_tendencia_exp_final <- tslm(passageiros_ts ~
trend, lambda = 0)

#sumario do modelo
summary(modelo_tendencia_exp_final)

#projeta os próximos 12 meses do futuro
modelo_tendencia_exp_final_proj <-
forecast(modelo_tendencia_exp_final, h=36, level=0.95)

#plota o grafico da serie temporal de treinamento e teste
plot(modelo_tendencia_exp_final_proj, xlab="Tempo",
ylab="Passageiros", ylim=c(1300, 2300), xlim=c(1991,
2007), bty="l", flty=2, main="Forecast from Exp
regression model")
axis(1, at=seq(1991, 2007, 1), labels=format(seq(1991,
2007,1)))
lines(modelo_tendencia_exp_final_proj$fitted, lwd=2,
col="blue")
```



Modelos de Regressão – Modelos com Sazonalidade

- Um padrão de sazonalidade na série temporal significa que observações que ocorrem em algumas épocas apresentam comportamentos consistentemente mais altos ou baixos daqueles que ocorrem em outras épocas.

- Exemplos são: comportamentos de dia da semana, mensais ou trimestrais. A série de passageiros de trem apresenta nítido padrões mensais (ggseasonplot):



Modelos de Regressão – Modelos com Sazonalidade

- A forma mais comum de capturar a sazonalidade num modelo de regressão é pela criação de uma variável categórica que define a sazonalidade de cada observação. Esta variável é então transformada em varáveis dummy (0 ou 1) e, por fim, incluída no modelo de regressão como variável explicativa.

Month	Ridership	Season	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov
Jan-91	1708.917	Jan	1	0	0	0	0	0	0	0	0	0	0
Feb-91	1620.586	Feb	0	1	0	0	0	0	0	0	0	0	0
Mar-91	1972.715	Mar	0	0	1	0	0	0	0	0	0	0	0
Apr-91	1811.665	Apr	0	0	0	1	0	0	0	0	0	0	0
May-91	1974.964	May	0	0	0	0	1	0	0	0	0	0	0
Jun-91	1862.356	Jun	0	0	0	0	0	1	0	0	0	0	0
Jul-91	1939.860	Jul	0	0	0	0	0	0	1	0	0	0	0
Aug-91	2013.264	Aug	0	0	0	0	0	0	0	1	0	0	0
Sep-91	1595.657	Sep	0	0	0	0	0	0	0	0	1	0	0
Oct-91	1724.924	Oct	0	0	0	0	0	0	0	0	0	1	0
Nov-91	1675.667	Nov	0	0	0	0	0	0	0	0	0	0	1
Dec-91	1813.863	Dec	0	0	0	0	0	0	0	0	0	0	0

- O modelo apresenta o seguinte formato:

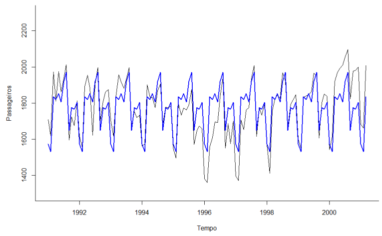
$$y_t = \beta_0 + \beta_1 Jan + \beta_2 Fev + \cdots + \beta_{11} Nov + e_t$$

- Com a equação acima podemos estimar 3 componentes da série temporal: nível (β_0), sazonalide (β_1 a β_{11}) e ruído (e_t), apenas a tendência não é modelada.

Modelos de Regressão – Modelos com Sazonalidade

- Calibramos o modelo de sazonalidade na série temporal de treinamento. A função tslm do R já cria as dummies automaticamente.

```
#Estima o modelo de tendência linear
modelo_sazonalidade_linear <- tslm(treinamento_ts ~ season)

#resumo do modelo
summary(modelo_sazonalidade_linear)

#plot modelo com sazonalidade
plot(treinamento_ts, xlab="Tempo", ylab="Passageiros", ylim=c(1300, 2300), bty="l")
lines(modelo_sazonalidade_linear$fitted.values, lwd=2)
```



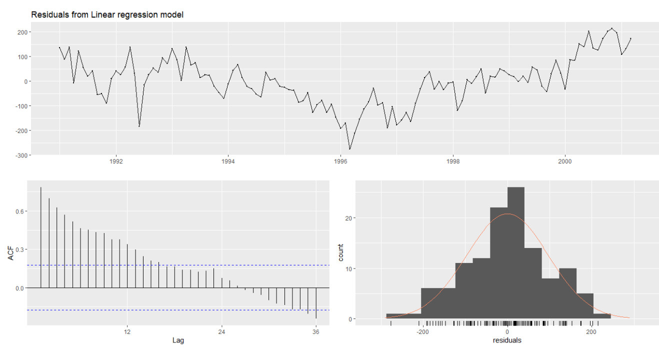

```
Call:
tslm(formula = treinamento_ts ~ season)

Residuals:
    Min      1Q  Median      3Q     Max
-276.165 -52.934   5.868  54.544 215.081

Coefficients:
           Estimate Std. Error t value Pr(>|t|)
(Intercept)  1573.97     30.58  51.475  < 2e-16 ***
season2       -42.93     43.24  -0.993   0.3230
season3       260.77     43.24   6.030 2.19e-08 ***
season4       245.09     44.31   5.531 2.14e-07 ***
season5       278.22     44.31   6.279 6.81e-09 ***
season6       233.46     44.31   5.269 6.82e-07 ***
season7       345.33     44.31   7.793 3.79e-12 ***
season8       396.66     44.31   8.952 9.19e-15 ***
season9        75.76     44.31   1.710   0.0901 .
season10      200.61     44.31   4.527 1.51e-05 ***
season11      192.36     44.31   4.341 3.14e-05 ***
season12      230.42     44.31   5.200 9.18e-07 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 101.4 on 111 degrees of freedom
Multiple R-squared:  0.6348,  Adjusted R-squared:  0.5986
F-statistic: 17.54 on 11 and 111 DF,  p-value: < 2.2e-16
```

Modelos de Regressão – Modelos com Tendência: Análise de Resíduos (Noise) – checkresiduals

- O R possui uma função chamada checkresiduals() que computa os valores de ACF e testes estatísticos de forma automática.

```
#verifica os resíduos com teste de Ljung-Box
checkresiduals(modelo_sazonalidade_linear, test="LB")
```

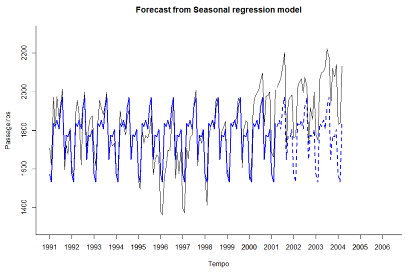
```
Ljung-Box test

data:  Residuals from Linear regression model
Q* = 481.74, df = 12, p-value < 2.2e-16

Model df: 12.   Total lags used: 24
```



Modelos de Regressão – Modelos com Sazonalidade

- Depois testamos o modelo na série temporal de validação

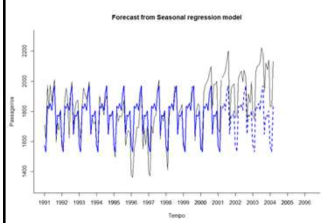
```
#projeta o modelo durante o período de validação
modelo_sazonalidade_linear_proj <-
forecast(modelo_sazonalidade_linear, h =
tam_amostra_teste, level=0)

#plota o grafico da serie temporal de treinamento e teste
plot(modelo_sazonalidade_linear_proj, xlab="Tempo",
ylab="Passageiros", xant="n" , ylim=c(1300, 2300),
xlim=c(1991, 2006.25), bty="l", flty=2, main="Forecast
from Seasonal regression model")

axis(1, at=seq(1991, 2006, 1), labels=format(seq(1991,
2006,1)))

lines(validacao_ts)
lines(modelo_sazonalidade_linear_proj$fitted, lwd=2,
col="blue")


#Verifica a acuracia do modelo
accuracy(modelo_sazonalidade_linear_proj,
validacao_ts)
```

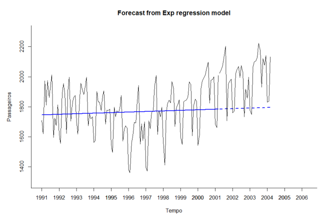


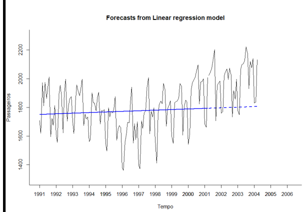
Modelo Sazonal					
	ME	RMSE	MAE	MPE	MAPE
Training set	0,0000	96,3404	75,1310	- 0,3099	4,3269
Test set	217,9267	229,6509	217,9267	10,8646	10,8646

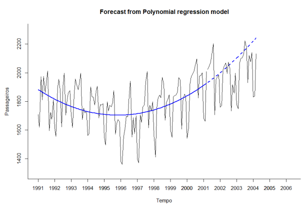
Modelos de Regressão – Modelos com Sazonalidade

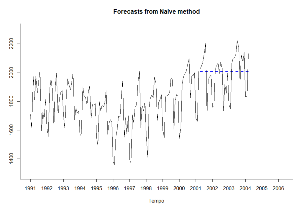
- Comparando com o Modelo Naive, Quadrático, Linear e Exponencial











Modelos de Regressão – Modelos com Sazonalidade

- Comparando com o Modelo Naive, Quadrático e Linear

Modelo Sazonal					
	ME	RMSE	MAE	MPE	MAPE
Training set	0,0000	96,3404	75,1310	- 0,3099	4,3269
Test set	217,9267	229,6509	217,9267	10,8646	10,8646

Modelo de Tendência Exponencial					
	ME	RMSE	MAE	MPE	MAPE
Training set	7,3750	159,0912	130,5631	- 0,4346	7,5540
Test set	203,2733	247,7550	216,2353	9,7213	10,4680

Modelo de Tendência Quadrático					
	ME	RMSE	MAE	MPE	MAPE
Training set	0,0000	146,9785	120,3980	- 0,7350	7,0158
Test set	- 83,9621	179,8494	133,7383	- 4,7254	7,0757

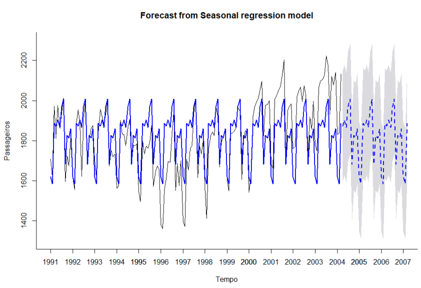
Modelo de Tendência Linear					
	ME	RMSE	MAE	MPE	MAPE
Training set	- 0,0000	158,9269	129,6778	- 0,8540	7,5360
Test set	193,1316	239,4863	209,4371	9,2099	10,1477

Modelo de Tendência Naive					
	ME	RMSE	MAE	MPE	MAPE
Training set	2,4509	168,1470	125,2975	- 0,3460	7,2714
Test set	- 14,7177	142,7551	115,9234	- 1,2750	6,0214

Modelos de Regressão – Modelos com Sazonalidade

- Projetar para os próximos 36 meses no futuro

```
#primeiramente reestimamos o modelo com todos os
dados de treinamento e validacao
modelo_sazonalidade_linear_final <- tslm(passageiros_ts
~ season)

#sumario do modelo
summary(modelo_sazonalidade_linear_final)

#projeta os próximos 12 meses do futuro
modelo_sazonalidade_linear_final_proj <-
forecast(modelo_sazonalidade_linear_final, h=36,
level=0.95)

#plota o grafico da serie temporal de treinamento e teste
plot(modelo_sazonalidade_linear_final_proj,
xlab="Tempo", ylab="Passageiros", ylim=c(1300, 2300),
xlim=c(1991, 2007), bty="l", flty=2,main="Forecast from
Seasonal regression model")
axis(1, at=seq(1991, 2007, 1), labels=format(seq(1991,
2007,1)))
lines(modelo_sazonalidade_linear_final_proj$fitted,
lwd=2, col="blue")
```



Modelos de Regressão – Modelos com Sazonalidade e Tendência

- Para a captura de todos os componentes da série temporal podemos combinar os modelos de tendência e sazonalidade:

Month	Ridership	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	t	t2
Jan-91	1708.917	1	0	0	0	0	0	0	0	0	0	0	1	1
Feb-91	1620.586	0	1	0	0	0	0	0	0	0	0	0	2	4
Mar-91	1972.715	0	0	1	0	0	0	0	0	0	0	0	3	9
Apr-91	1811.665	0	0	0	1	0	0	0	0	0	0	0	4	16
May-91	1974.964	0	0	0	0	1	0	0	0	0	0	0	5	25
Jun-91	1862.356	0	0	0	0	0	1	0	0	0	0	0	6	36
Jul-91	1939.860	0	0	0	0	0	0	1	0	0	0	0	7	49
Aug-91	2013.264	0	0	0	0	0	0	0	1	0	0	0	8	64
Sep-91	1595.657	0	0	0	0	0	0	0	0	1	0	0	9	81
Oct-91	1724.924	0	0	0	0	0	0	0	0	0	1	0	10	100
Nov-91	1675.667	0	0	0	0	0	0	0	0	0	0	1	11	121
Dec-91	1813.863	0	0	0	0	0	0	0	0	0	0	0	12	144

- O modelo apresenta o seguinte formato:

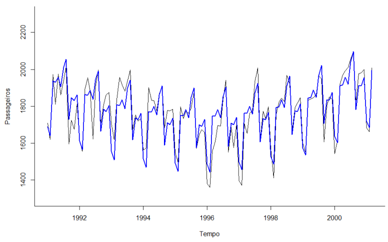
$$y_t = \beta_0 + \beta_1 Jan + \beta_2 Fev + \cdots + \beta_{11} Nov + \beta_{12} t + \beta_{13} t^2 + e_t$$

- Com a equação acima podemos estimar 3 componentes da série temporal: nível (β_0), sazonalidade (β_1 a β_{11}) , tendência (β_{12} a β_{13}) e ruído (e_t).

Modelos de Regressão – Modelos com Sazonalidade e Tendência

- Calibramos o modelo de sazonalidade na série temporal de treinamento

```
#Estima o modelo de tendência linear
modelo_sazonal_tend_linear <- tslm(treinamento_ts ~
season + trend + I(trend^2))

#resumo do modelo
summary(modelo_sazonal_tend_linear)

#plot modelo com sazonal_tend
plot(treinamento_ts, xlab="Tempo", ylab="Passageiros",
ylim=c(1300, 2300), bty="l")
lines(modelo_sazonal_tend_linear$fitted.values, lwd=2,
col="blue")
```



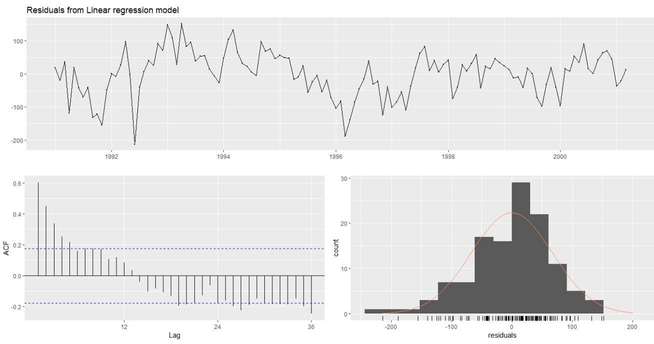

```
Call:
tslm(formula = treinamento_ts ~ season + trend + I(trend^2))

Residuals:
     Min       1Q   Median       3Q      Max
-213.775  -39.363    9.711   42.422  152.187

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  1.697e+03  2.768e+01  61.318  < 2e-16 ***
season2     -4.325e+01  3.024e+01  -1.430  0.15556
season3      2.600e+02  3.024e+01   8.598 6.60e-14 ***
season4      2.606e+02  3.102e+01   8.401 1.83e-13 ***
season5      2.938e+02  3.102e+01   9.471 6.89e-16 ***
season6      2.490e+02  3.102e+01   8.026 1.26e-12 ***
season7      3.606e+02  3.102e+01  11.626  < 2e-16 ***
season8      4.117e+02  3.102e+01  13.270  < 2e-16 ***
season9      9.032e+01  3.102e+01   2.911  0.00437 **
season10     2.146e+02  3.102e+01   6.917 3.29e-10 ***
season11     2.057e+02  3.103e+01   6.629 1.34e-09 ***
season12     2.429e+02  3.103e+01   7.829 3.44e-12 ***
trend       -7.156e+00  7.293e-01  -9.812  < 2e-16 ***
I(trend^2)   6.074e-02  5.698e-03  10.660  < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 70.92 on 109 degrees of freedom
Multiple R-squared:  0.8246,  Adjusted R-squared:  0.8037
F-statistic: 39.42 on 13 and 109 DF,  p-value: < 2.2e-16
```

Modelos de Regressão – Modelos com Tendência: Análise de Resíduos (Noise) – checkresiduals

- O R possui uma função chamada checkresiduals() que computa os valores de ACF e testes estatísticos de forma automática.

```
#verifica os resíduos com teste de Ljung-Box
checkresiduals(modelo_sazonal_tend_linear, test="LB")
```

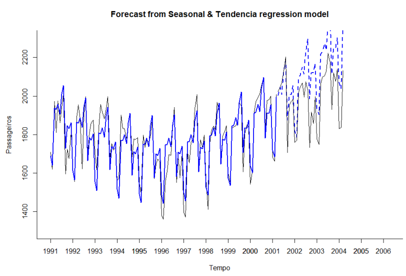
```
Ljung-Box test

data:  Residuals from Linear regression model
Q* = 151.62, df = 10, p-value < 2.2e-16

Model df: 14.   Total lags used: 24
```



Modelos de Regressão – Modelos com Sazonalidade

- Depois testamos o modelo na série temporal de validação

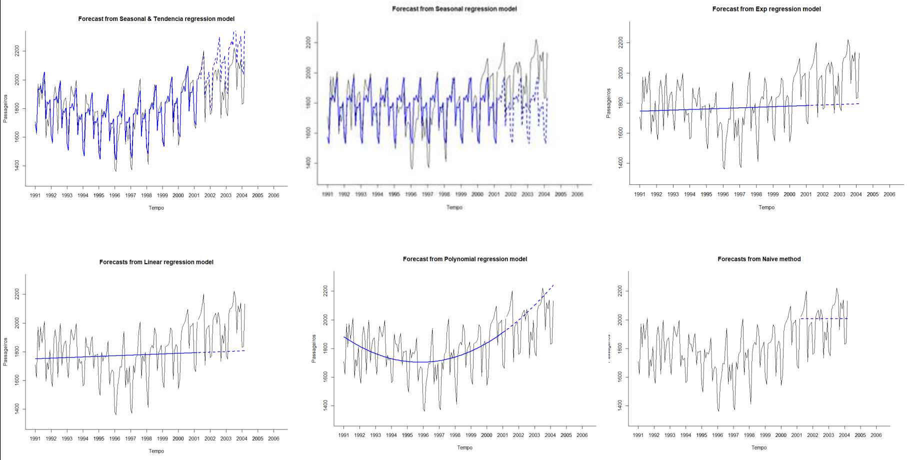
```
#projeta o modelo durante o período de validação
modelo_sazonal_tend_linear_proj <-
forecast(modelo_sazonal_tend_linear, h =
tam_amostra_teste, level=0)

#plota o grafico da serie temporal de treinamento e teste
plot(modelo_sazonal_tend_linear_proj, xlab="Tempo",
ylab="Passageiros", xant="n" , ylim=c(1300, 2300),
xlim=c(1991, 2006.25), bty="l", flty=2, main="Forecast
from Seasonal & Tendencia regression model")

axis(1, at=seq(1991, 2006, 1), labels=format(seq(1991,
2006,1)))

lines(validacao_ts)
lines(modelo_sazonal_tend_linear_proj$fitted, lwd=2,
col="blue")


#Verifica a acuracia do modelo
accuracy(modelo_sazonal_tend_linear_proj,
validacao_ts)
```



Modelo Com Sazonalidade e Tendência Quadrática					
	ME	RMSE	MAE	MPE	MAPE
Training set	0,0000	66,7614	51,9509	- 0,1526	3,0155
Test set	- 126,1654	153,2507	131,7250	- 6,4315	6,6987

Modelos de Regressão – Modelos com Sazonalidade

- Comparando com o Modelo Naive, Quadrático, Linear, Exponencial e Sazonal



Modelos de Regressão – Modelos com Sazonalidade

- Comparando com o Modelo Naive, Quadrático e Linear

Modelo Com Sazonalidade e Tendência Quadrática					
	ME	RMSE	MAE	MPE	MAPE
Training set	0,0000	66,7614	51,9509	- 0,1526	3,0155
Test set	- 126,1654	153,2507	131,7250	- 6,4315	6,6987

Modelo Sazonal					
	ME	RMSE	MAE	MPE	MAPE
Training set	0,0000	96,3404	75,1310	- 0,3099	4,3269
Test set	217,9267	229,6509	217,9267	10,8646	10,8646

Modelo de Tendência Exponencial					
	ME	RMSE	MAE	MPE	MAPE
Training set	7,3750	159,0912	130,5631	- 0,4346	7,5540
Test set	203,2733	247,7550	216,2353	9,7213	10,4680

Modelo de Tendência Quadrático					
	ME	RMSE	MAE	MPE	MAPE
Training set	0,0000	146,9785	120,3980	- 0,7350	7,0158
Test set	- 83,9621	179,8494	133,7383	- 4,7254	7,0757

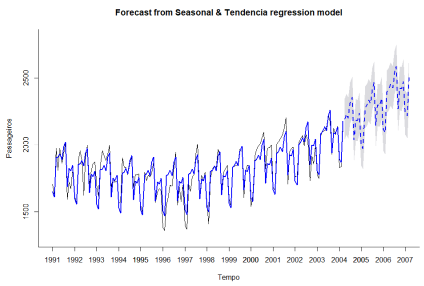
Modelo de Tendência Linear					
	ME	RMSE	MAE	MPE	MAPE
Training set	- 0,0000	158,9269	129,6778	- 0,8540	7,5360
Test set	193,1316	239,4863	209,4371	9,2099	10,1477

Modelo de Tendência Naive					
	ME	RMSE	MAE	MPE	MAPE
Training set	2,4509	168,1470	125,2975	- 0,3460	7,2714
Test set	- 14,7177	142,7551	115,9234	- 1,2750	6,0214

Modelos de Regressão – Modelos com Sazonalidade

- Projetar para os próximos 36 meses no futuro

```
#primeiramente reestimamos o modelo com todos os
dados de treinamento e validacao
modelo_sazonalidade_linear_final <- tslm(passageiros_ts
~ season)

#sumario do modelo
summary(modelo_sazonalidade_linear_final)

#projeta os próximos 12 meses do futuro
modelo_sazonalidade_linear_final_proj <-
forecast(modelo_sazonalidade_linear_final, h=36,
level=0.95)

#plota o grafico da serie temporal de treinamento e teste
plot(modelo_sazonalidade_linear_final_proj,
xlab="Tempo", ylab="Passageiros", ylim=c(1300, 2300),
xlim=c(1991, 2007), bty="l", flty=2,main="Forecast from
Seasonal regression model")
axis(1, at=seq(1991, 2007, 1), labels=format(seq(1991,
2007,1)))
lines(modelo_sazonalidade_linear_final_proj$fitted,
lwd=2, col="blue")
```



Sumário

- Apresentações
- Regras do Jogo
- Introdução as Séries Temporais e Projeção
- Dados no contexto de Séries Temporais
- Avaliação de Performance
- Métodos de Projeção: Visão Geral
- Métodos de Regressão
- **Métodos de Projeção: Suavização**
- Modelos de Regressão ARIMA

Modelos de Suavização – Introdução

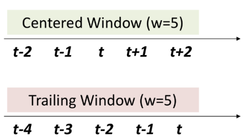
- Modelos de suavização são baseados na utilização da média de comportamento da série temporal durante vários períodos com o objetivo de reduzir ruídos (noise);
- Os dois modelos mais simples de suavização são médias móveis e suavização exponencial, que são indicados para projeção de séries temporais sem tendência ou sazonalidade;
- Em ambos modelos as projeções são médias de valores passados das séries;
- Modelos de suavização são orientados a dados e flexíveis o bastante para se adaptar a mudanças nos padrões da séries temporais ao longo do tempo;
- Como são modelos que aprendem com os dados necessitam de séries temporais mais longas;
- Modelos aprendidos dos dados podem ser extremamente complexos (caixas pretas);

Modelos de Suavização – Média Móvel – Moving Average (MA)

- O modelo de média móvel é um suavizador simples que consiste na obtenção da média de valores dentro de uma janela de períodos consecutivos, gerando assim uma série temporal de médias;
- Uma média móvel com janela de largura w significa calcular a média de cada conjunto de w valores consecutivos, onde w é definido pelo usuário;
- Em geral existem 2 tipos de médias móveis:

  1. Média móvel centralizada (Centered Window): normalmente utilizada para a visualização de tendências, pois o processo de calculo da média remove sazonalidades e ruídos dos dados, tornado a visualização da tendência mais fácil;

  2. Média móvel simples (Trailing Window): normalmente utilizadas para projeção, onde a janela de médias é calculada do momento atual para os valores mais antigos.



Modelos de Suavização – Média Móvel Centralizada

FGV

- Na média móvel centralizada, o valor da média móvel no momento t (MA_t) é calculado pela centralização da janela no tempo t e calculando a média para todos os valores da janela w.

$$MA_t = \frac{\left(y_{t-(w-1)/2} + \cdots + y_{t-1} + y_t + y_{t+1} + \cdots + y_{t+(w-1)/2}\right)}{w}$$

- Por exemplo, com uma janela de largura $w = 5$, a média móvel no ponto t=3 significa calcular a média dos pontos 1,2,3,4 e 5.
- A escolha da largura da janela numa série com sazonalidade é simples, pois o objetivo é suprimir o efeito da sazonalidade para melhor visualizar a tendência.
- No caso do exemplo de passageiros, conforme verificado anteriormente existe uma sazonalidade mensal e, portanto, uma janelas $w = 12$ seria justificável.
- Como a média móvel centralizada utiliza valores do futuro, o seu uso para projeção é muito limitado.

Modelos de Suavização – Média Móvel Simples

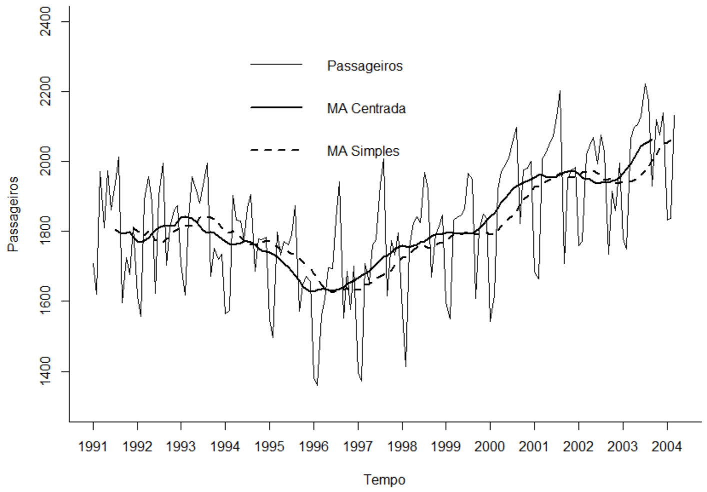
FGV

- Na média móvel simples, o valor da média móvel no momento t (MA_t) é calculado pela centralização da janela no tempo t e calculando a média para todos os valores da janela w. Utilizando sempre os valores mais recentes.

$$MA_t = \frac{(y_t + y_{t-1} + \cdots + y_{t-w+1})}{w}$$

- Por exemplo, com uma janela de largura $w = 5$, a média móvel no ponto t=10 significa calcular a média dos pontos 10, 9,8,7,6.

Modelos de Suavização – Média Móvel Simples



Modelos de Suavização – Média Móvel Simples - Projeção

- A projeção de k-passos a no futuro F_{t+k} $(k = 1,2,3, \ldots)$ é a média dos valores mais recentes dentro da janela w.

$$F_{t+k} = \frac{(y_t + y_{t-1} + \cdots + y_{t-w+1})}{w}$$

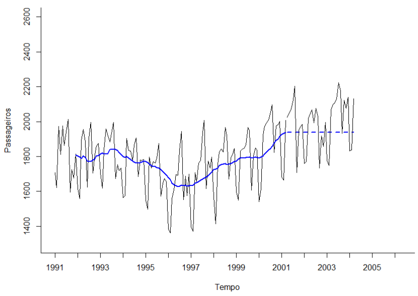
- Por exemplo, na série de passageiros, para projetar o volume de fevereiro 1992 em diante, dada a informação disponível até janeiro de 1992 e utilizando uma janela móvel de $w = 12$, nos utilizaríamos como projeção a média dos 12 meses mais recentes (fevereiro 1991 até janeiro 1992).

- O tamanho da janela w deverá ser definido pelo usuário, lembrando que uma janela pequena pode levar ao overfitting dos dados e uma janela grande ao underfitting. O modelo de projeção naive usa um w=1.

Modelos de Suavização – Média Móvel Simples - Projeção

- Utilizando o mesmo exemplo dos exercícios anteriores vamos estimar a média móvel para período de treinamento e projetar para o período de validação.

```
#estima o modelo de MA na base de treinamento
ma_simples <- rollmean(treinamento_ts, k=12, align="right")

#obtem a média da última janela movel de 12 meses para projeção
ultima_ma <- tail(ma_simples, 1)

#cria uma projeção que é a repeticao da ultima media da janela
para o periodo de validacao
ma_simples_proj <- ts(rep(ultima_ma, tam_amostra_teste),
start=c(1991, tam_amostra_treinamento+1), end = c(1991,
tam_amostra_treinamento + tam_amostra_teste), freq=12)
```

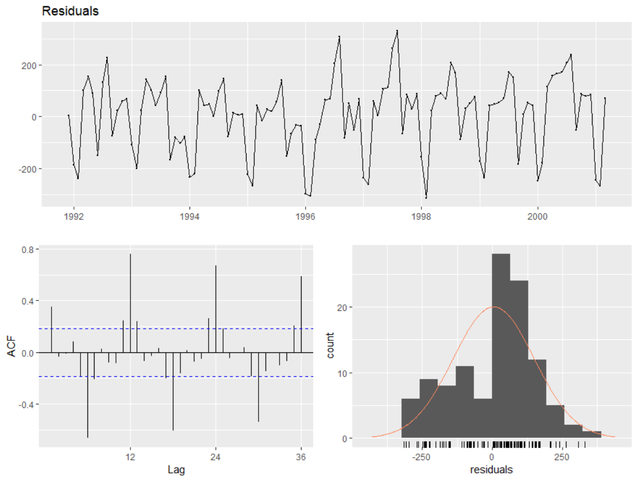


Modelo de Média Móvel MA(12)					
	ME	RMSE	MAE	MPE	MAPE
Training set	4,7570	143,8445	117,0441	- 0,4299	6,8130
Test set	54,7296	152,1767	135,1550	2,2278	6,7771

- Note que para os 12 primeiros meses do período de teste não existe uma estimação, pois trata-se da janela de estimação.
- O valor projetado para o período de validação é o mesmo (1.938,481), pois o método assume que a informação existe apenas até março de 2001.

Modelos de Regressão – Modelos de Suavização – checkresiduals

- O R possui uma função chamada checkresiduals() que computa os valores de ACF e testes estatísticos de forma automática.



Modelos de Regressão – Modelos de Suavização

- Comparando com o Modelo Naive, Quadrático e Linear

Modelo de Média Móvel MA(12)

	ME	RMSE	MAE	MPE	MAPE
Training set	4,7570	143,8445	117,0441	- 0,4299	6,8130
Test set	54,7296	152,1767	135,1550	2,2278	6,7771

Modelo Com Sazonalidade e Tendência Quadrática

	ME	RMSE	MAE	MPE	MAPE
Training set	0,0000	66,7614	51,9509	- 0,1526	3,0155
Test set	- 126,1654	153,2507	131,7250	- 6,4315	6,6987

Modelo Sazonal

	ME	RMSE	MAE	MPE	MAPE
Training set	0,0000	96,3404	75,1310	- 0,3099	4,3269
Test set	217,9267	229,6509	217,9267	10,8646	10,8646

Modelo de Tendência Exponencial

	ME	RMSE	MAE	MPE	MAPE
Training set	7,3750	159,0912	130,5631	- 0,4346	7,5540
Test set	203,2733	247,7550	216,2353	9,7213	10,4680

Modelo de Tendência Quadrático

	ME	RMSE	MAE	MPE	MAPE
Training set	0,0000	146,9785	120,3980	- 0,7350	7,0158
Test set	- 83,9621	179,8494	133,7383	- 4,7254	7,0757

Modelo de Tendência Linear

	ME	RMSE	MAE	MPE	MAPE
Training set	- 0,0000	158,9269	129,6778	- 0,8540	7,5360
Test set	193,1316	239,4863	209,4371	9,2099	10,1477

Modelo de Tendência Naive

	ME	RMSE	MAE	MPE	MAPE
Training set	2,4509	168,1470	125,2975	- 0,3460	7,2714
Test set	- 14,7177	142,7551	115,9234	- 1,2750	6,0214

- Nitidamente o modelo de MA não é adequado na para séries com presença de tendência e sazonalidade.
- Formas de corrigir estes problemas normalmente são utilizados modelos de regressão, modelos de suavização exponencial e diferenciação.

Modelos de Suavização – Suavização Exponencial Simples

- Um método de projeção popular é o modelo de suavização exponencial.
- O método de suavização exponencial é similar o projeção com médias móveis, diferindo apenas no forma como o valor médio é calculado. Ao invés de uma média simples para todos os valores da janela w, o modelo de suavização exponencial tira uma média ponderada dos valores da janela. Desta forma os valores mais recentes possuem um peso maior:

$$F_{t+1} = \alpha y_t + \alpha(1-\alpha)y_{t-1} + \alpha(1-\alpha)^2 y_{t-2} \ldots$$

- Onde α é uma constante entre 0 e 1, chamada de constante de suavização. Quanto mais próximo de 1 maior o peso do passado recente e menor o peso do passado distante. Para valores próximos de 0 o pesos tente ao valor da média simples onde todos os pontos possuem o mesmo peso;
- Valores de α são normalmente definidos entre 0.1 e 0.2, mas o usuário pode investigar o melhor ajuste. Sempre atento para as questões de overfitting/underfitting.
- O modelo de suavização exponencial também pode ser representado da seguinte forma:

$$F_{t+1} = F_t + \alpha e_t$$

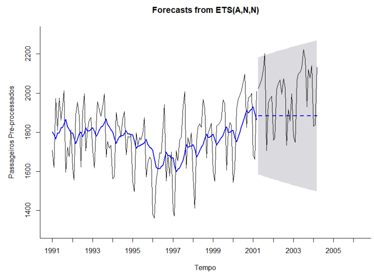
- Assim como no caso da média móvel as projeções k-períodos a frente são dadas pela última projeção disponível:

$$F_{t+k} = F_{t+1}$$

Modelos de Suavização – Suavização Exponencial Simples - Projeção

- Utilizando o mesmo exemplo dos exercícios anteriores vamos estimar a suavização exponencial simples para o período de treinamento e projetar para o período de validação. O R possui a função **ets** para este tipo de modelo.

```
#estima o modelo de suavizacao na base de treinamento
modelo_ses <- ets(treinamento_ts, model = "ANN")

#resumo modelo
summary(modelo_ses)

#projeta os proximos 12 meses
modelo_ses_proj <- forecast(modelo_ses, h=tam_amostra_teste,
level=0.95)
```

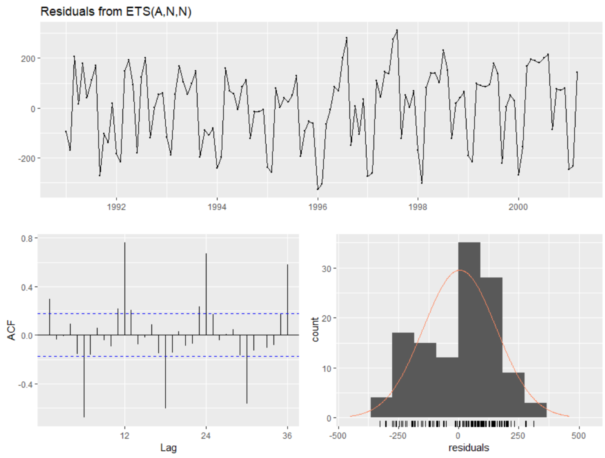



```
ETS(A,N,N)

Call:
 ets(y = treinamento_ts, model = "ANN")

  Smoothing parameters:
    alpha = 0.1382
```

- $F_{t+1} = F_t + \alpha e_t = \alpha y_t + (1-\alpha)\, F_t$
- Pois assumimos para o futuro que $e_t = 0$.
- O valor projetado para o período de validação é o mesmo (1884,196), pois o método assume que a informação existe apenas até março de 2001.
- $1884{,}19 = 0.1382 \times 2007.92 + (1 - 0.1382) \times 1864.35$

Modelos de Regressão – Modelos de Suavização – checkresiduals

- O R possui uma função chamada checkresiduals() que computa os valores de ACF e testes estatísticos de forma automática.



Modelos de Regressão – Modelos de Suavização

FGV

- Comparando

Modelo de Suavização sem tendência e sazonalidade					
	ME	RMSE	MAE	MPE	MAPE
Training set	4,8306	151,3034	127,9211	- 0,4566	7,4106
Test set	109,0143	179,0154	160,6982	4,9657	7,9065

Modelo de Média Móvel MA(12)					
	ME	RMSE	MAE	MPE	MAPE
Training set	4,7570	143,8445	117,0441	- 0,4299	6,8130
Test set	54,7296	152,1767	135,1550	2,2278	6,7771

Modelo Com Sazonalidade e Tendência Quadrática					
	ME	RMSE	MAE	MPE	MAPE
Training set	0,0000	66,7614	51,9509	- 0,1526	3,0155
Test set	- 126,1654	153,2507	131,7250	- 6,4315	6,6987

Modelo Sazonal					
	ME	RMSE	MAE	MPE	MAPE
Training set	0,0000	96,3404	75,1310	- 0,3099	4,3269
Test set	217,9267	229,6509	217,9267	10,8646	10,8646

Modelo de Tendência Exponencial					
	ME	RMSE	MAE	MPE	MAPE
Training set	7,3750	159,0912	130,5631	- 0,4346	7,5540
Test set	203,2733	247,7550	216,2353	9,7213	10,4680

Modelo de Tendência Quadrático					
	ME	RMSE	MAE	MPE	MAPE
Training set	0,0000	146,9785	120,3980	- 0,7350	7,0158
Test set	- 83,9621	179,8494	133,7383	- 4,7254	7,0757

Modelo de Tendência Linear					
	ME	RMSE	MAE	MPE	MAPE
Training set	- 0,0000	158,9269	129,6778	- 0,8540	7,5360
Test set	193,1316	239,4863	209,4371	9,2099	10,1477

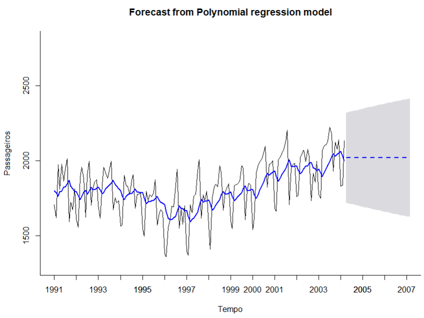
Modelo de Tendência Naive					
	ME	RMSE	MAE	MPE	MAPE
Training set	2,4509	168,1470	125,2975	- 0,3460	7,2714
Test set	- 14,7177	142,7551	115,9234	- 1,2750	6,0214

Modelos de Regressão – Modelos de Suavização Exponencial Simples

FGV

- Projetar para os próximos 36 meses no futuro

```
#primeiramente reestimamos o modelo com todos os
dados de treinamento e validacao
modelo_ses_final <- ets(passageiros_ts, model = "ANN")

#sumario do modelo
summary(modelo_ses_final)

#projeta os próximos 36 meses do futuro
modelo_ses_final_proj <- forecast(modelo_ses_final,
h=36, level=0.95)

#plota o grafico da serie temporal de treinamento e teste
plot(modelo_ses_final_proj, xlab="Tempo",
ylab="Passageiros", ylim=c(1300, 2800), xlim=c(1991,
2007), bty="l", flty=2, main="Forecast from Polynomial
regression model")
axis(1, at=seq(1991, 2007, 1), labels=format(seq(1991,
2007,1)))
lines(modelo_ses_final_proj$fitted, lwd=2, col="blue")
```



Modelos de Suavização – Suavização Exponencial Modelos complexos

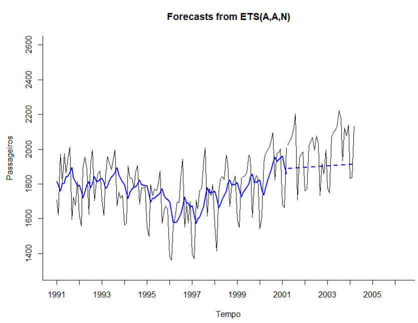
Para endereçar as limitações de sazonalidade e tendência existem algumas combinações de modelos exponenciais:

Modelo	Equação	Significado
Séries com tendência aditiva (Holt)	$F_{t+k} = L_t + kT_t$	Permite a tendência mudar por um valor constante de tempos em tempos, ao invés de permanecer constante como no modelo de regressão linear. (model="AAN")
Séries com tendência multiplicativa	$F_{t+k} = L_t \times T_t^k$	Permite a tendência mudar por um valor percentual de tempos em tempos, ao invés de permanecer constante como no modelo de regressão linear. (model="AMN")
Séries com tendência e sazonalidade aditiva (Holt-Winter)	$F_{t+k} = L_t + kT_t + S_{t+k-M}$	Modelo que considera tanto a tendência e sazonalidade mudando ao longo do tempo por quantidades constantes e erro multiplicativo. (model="MAA")
Séries com tendência aditiva e sazonalidade multiplicativa (Holt-Winter)	$F_{t+k} = (L_t + kT_t)S_{t+k-M}$	Modelo que considera tanto a tendência e sazonalidade mudando ao longo do tempo por quantidades constante e percentual respectivamente e erro multiplicativo. (model="MAM")

Modelos de Suavização – Suavização Exponencial Modelos Complexos: Série com tendência aditiva (Holt)

- Utilizando o mesmo exemplo dos exercícios anteriores vamos estimar o modelo exp para o período de treinamento e projetar para o período de validação.

```
#estima o modelo de suavizacao na base de treinamento
modelo_ses <- ets(treinamento_ts, model = "AAN")

#resumo modelo
summary(modelo_ses)

#projeta os proximos 12 meses
modelo_ses_proj <- forecast(modelo_ses, h=tam_amostra_teste,
level=0.95)
```

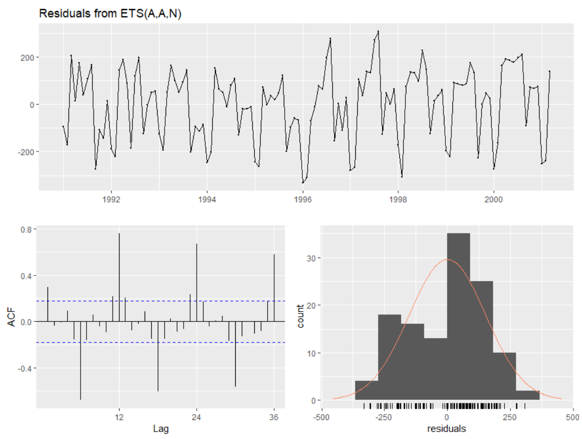



```
Call:
 ets(y = treinamento_ts, model = "AAN")

  Smoothing parameters:
    alpha = 0.137
    beta  = 1e-04

  Initial states:
    l = 1801.8411
    b = 0.7167
```

- $F_{t+k} = L_t + kT_t$
- $L_t = \alpha y_t + (1-\alpha)(L_{t-1} + T_{t-1})$
- $T_t = \beta(L_t - L_{t-1}) + (1-\beta)T_{t-1}$
- Note que a projeção já apresenta uma tendência e os valores não são mais iguais para todos os períodos.

Modelos de Regressão – Modelos de Suavização – checkresiduals

- O R possui uma função chamada checkresiduals() que computa os valores de ACF e testes estatísticos de forma automática.



Modelos de Regressão – Modelos de Suavização

- Comparando

Modelo de Suavização Exponencial com Tendência Aditiva (AAN)					
	ME	RMSE	MAE	MPE	MAPE
Training set	0,2969	151,2727	127,0256	- 0,7150	7,3778
Test set	91,3318	168,2559	152,3572	4,0770	7,5391

Modelo de Suavização sem tendência e sazonalidade					
	ME	RMSE	MAE	MPE	MAPE
Training set	4,8306	151,3034	127,9211	- 0,4566	7,4106
Test set	109,0143	179,0154	160,6982	4,9657	7,9065

Modelo de Média Móvel MA(12)					
	ME	RMSE	MAE	MPE	MAPE
Training set	4,7570	143,8445	117,0441	- 0,4299	6,8130
Test set	54,7296	152,1767	135,1550	2,2278	6,7771

Modelo Com Sazonalidade e Tendência Quadrática					
	ME	RMSE	MAE	MPE	MAPE
Training set	0,0000	66,7614	51,9509	- 0,1526	3,0155
Test set	- 126,1654	153,2507	131,7250	- 6,4315	6,6987

Modelo Sazonal					
	ME	RMSE	MAE	MPE	MAPE
Training set	0,0000	96,3404	75,1310	- 0,3099	4,3269
Test set	217,9267	229,6509	217,9267	10,8646	10,8646

Modelo de Tendência Exponencial					
	ME	RMSE	MAE	MPE	MAPE
Training set	7,3750	159,0912	130,5631	- 0,4346	7,5540
Test set	203,2733	247,7550	216,2353	9,7213	10,4680

Modelo de Tendência Quadrático					
	ME	RMSE	MAE	MPE	MAPE
Training set	0,0000	146,9785	120,3980	- 0,7350	7,0158
Test set	- 83,9621	179,8494	133,7383	- 4,7254	7,0757

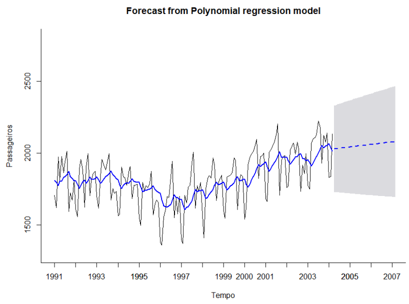
Modelo de Tendência Linear					
	ME	RMSE	MAE	MPE	MAPE
Training set	- 0,0000	158,9269	129,6778	- 0,8540	7,5360
Test set	193,1316	239,4863	209,4371	9,2099	10,1477

Modelo de Tendência Naive					
	ME	RMSE	MAE	MPE	MAPE
Training set	2,4509	168,1470	125,2975	- 0,3460	7,2714
Test set	- 14,7177	142,7551	115,9234	- 1,2750	6,0214

Modelos de Regressão – Modelos de Suavização

- Projetar para os próximos 36 meses no futuro

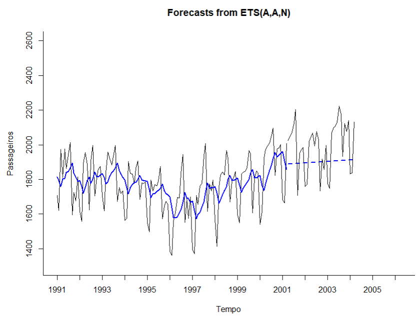
```
#primeiramente reestimamos o modelo com todos os dados de treinamento e validacao
modelo_ses_final <- ets(passageiros_ts, model = "AAN")

#sumario do modelo
summary(modelo_ses_final)

#projeta os próximos 36 meses do futuro
modelo_ses_final_proj <- forecast(modelo_ses_final, h=36, level=0.95)

#plota o grafico da serie temporal de treinamento e teste
plot(modelo_ses_final_proj, xlab="Tempo", ylab="Passageiros", ylim=c(1300, 2800), xlim=c(1991, 2007), bty="l", flty=2, main="Forecast from Polynomial regression model")
axis(1, at=seq(1991, 2007, 1), labels=format(seq(1991, 2007,1)))
lines(modelo_ses_final_proj$fitted, lwd=2, col="blue")
```



Modelos de Suavização – Suavização Exponencial Modelos Complexos: Série com tendência multiplicativa (Holt)

- Utilizando o mesmo exemplo dos exercícios anteriores vamos estimar o modelo exp para o período de treinamento e projetar para o período de validação.

```
#estima o modelo de suavizacao na base de treinamento
modelo_ses <- ets(treinamento_ts, model = "MMN")

#resumo modelo
summary(modelo_ses)

#projeta os proximos 12 meses
modelo_ses_proj <- forecast(modelo_ses, h=tam_amostra_teste, level=0.95)
```

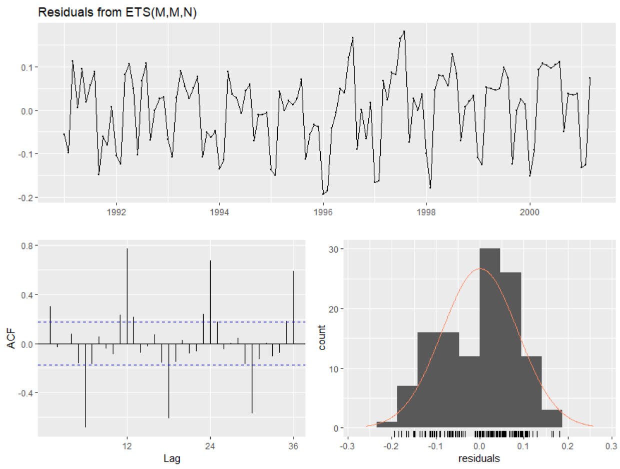



```
ETS(M,M,N)

Call:
 ets(y = treinamento_ts, model = "MMN")

  Smoothing parameters:
    alpha = 0.1389
    beta  = 1e-04

  Initial states:
    l = 1810.2189
    b = 1.0004
```

- $F_{t+k} = L_t + T_t^k$
- $L_t = \alpha y_t + (1-\alpha)(L_{t-1} \times T_{t-1})$
- $T_t = \beta\left(\frac{L_t}{L_{t-1}}\right) + (1-\beta)T_{t-1}$
- Note que a projeção já apresenta uma tendência e os valores não são mais iguais para todos os períodos.

Modelos de Regressão – Modelos de Suavização – checkresiduals

- O R possui uma função chamada checkresiduals() que computa os valores de ACF e testes estatísticos de forma automática.



Modelos de Regressão – Modelos de Suavização

- Comparando

Modelo de Suavização Exponencial com Tendência Multiplicativa (MMN)

	ME	RMSE	MAE	MPE	MAPE
Training set	- 0,6596	151,3118	126,9160	- 0,7679	7,3756
Test set	88,1354	166,4706	150,8469	3,9163	7,4725

Modelo de Suavização Exponencial com Tendência Aditiva (AAN)

	ME	RMSE	MAE	MPE	MAPE
Training set	0,2969	151,2727	127,0256	- 0,7150	7,3778
Test set	91,3318	168,2559	152,3572	4,0770	7,5391

Modelo de Suavização sem tendência e sazonalidade

	ME	RMSE	MAE	MPE	MAPE
Training set	4,8306	151,3034	127,9211	- 0,4566	7,4106
Test set	109,0143	179,0154	160,6982	4,9657	7,9065

Modelo de Média Móvel MA(12)

	ME	RMSE	MAE	MPE	MAPE
Training set	4,7570	143,8445	117,0441	- 0,4299	6,8130
Test set	54,7296	152,1767	135,1550	2,2278	6,7771

Modelo Com Sazonalidade e Tendência Quadrática

	ME	RMSE	MAE	MPE	MAPE
Training set	0,0000	66,7614	51,9509	- 0,1526	3,0155
Test set	- 126,1654	153,2507	131,7250	- 6,4315	6,6987

Modelo Sazonal

	ME	RMSE	MAE	MPE	MAPE
Training set	0,0000	96,3404	75,1310	- 0,3099	4,3269
Test set	217,9267	229,6509	217,9267	10,8646	10,8646

Modelo de Tendência Exponencial

	ME	RMSE	MAE	MPE	MAPE
Training set	7,3750	159,0912	130,5631	- 0,4346	7,5540
Test set	203,2733	247,7550	216,2353	9,7213	10,4680

Modelo de Tendência Quadrático

	ME	RMSE	MAE	MPE	MAPE
Training set	0,0000	146,9785	120,3980	- 0,7350	7,0158
Test set	- 83,9621	179,8494	133,7383	- 4,7254	7,0757

Modelo de Tendência Linear

	ME	RMSE	MAE	MPE	MAPE
Training set	- 0,0000	158,9269	129,6778	- 0,8540	7,5360
Test set	193,1316	239,4863	209,4371	9,2099	10,1477

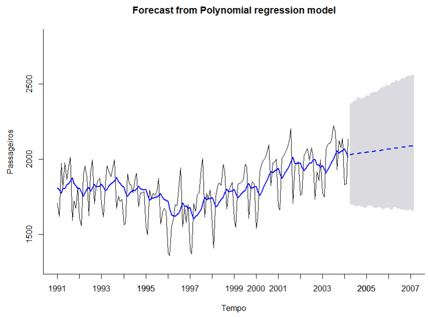
Modelo de Tendência Naive

	ME	RMSE	MAE	MPE	MAPE
Training set	2,4509	168,1470	125,2975	- 0,3460	7,2714
Test set	- 14,7177	142,7551	115,9234	- 1,2750	6,0214

Modelos de Regressão – Modelos de Suavização

- Projetar para os próximos 36 meses no futuro

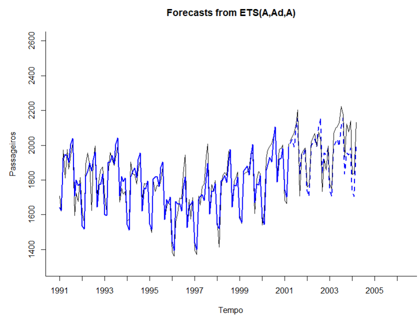
```
#primeiramente reestimamos o modelo com todos os
dados de treinamento e validacao
modelo_ses_final <- ets(passageiros_ts, model =
"MMN")

#sumario do modelo
summary(modelo_ses_final)

#projeta os próximos 36 meses do futuro
modelo_ses_final_proj <- forecast(modelo_ses_final,
h=36, level=0.95)

#plota o grafico da serie temporal de treinamento e teste
plot(modelo_ses_final_proj, xlab="Tempo",
ylab="Passageiros", ylim=c(1300, 2800), xlim=c(1991,
2007), bty="l", flty=2, main="Forecast from Polynomial
regression model")
axis(1, at=seq(1991, 2007, 1), labels=format(seq(1991,
2007,1)))
lines(modelo_ses_final_proj$fitted, lwd=2, col="blue")
```



Modelos de Suavização – Suavização Exponencial Modelos Complexos: Série com tendência e sazonalidade aditiva (Holt-Winter)

- Utilizando o mesmo exemplo dos exercícios anteriores vamos estimar o modelo exp para o período de treinamento e projetar para o período de validação.

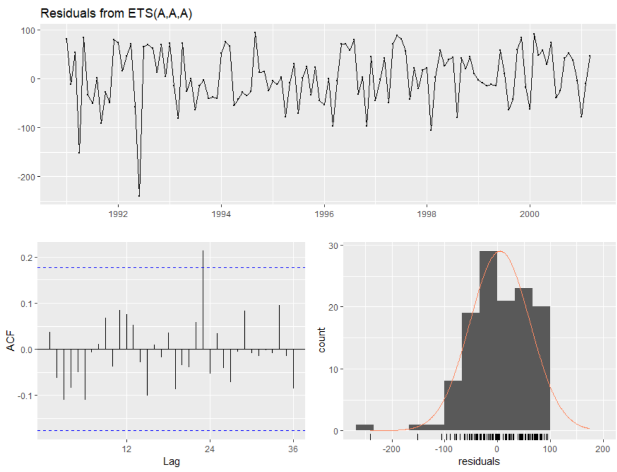
```
#estima o modelo de suavizacao na base de treinamento
modelo_ses <- ets(treinamento_ts, model = "AAA")

#resumo modelo
summary(modelo_ses)

#projeta os proximos 12 meses
modelo_ses_proj <- forecast(modelo_ses, h=tam_amostra_teste,
level=0.95)


ETS(A,A,A)

Call:
 ets(y = treinamento_ts, model = "AAA")

  Smoothing parameters:
    alpha = 0.5817
    beta  = 1e-04
    gamma = 1e-04
```



- $F_{t+k} = L_t + kT_t + S_{t+k-M}$
- $L_t = \alpha(y_t - S_{t-M}) + (1-\alpha)(L_{t-1} + T_{t-1})$
- $T_t = \beta(L_t - L_{t-1}) + (1-\beta)T_{t-1}$
- $S_t = \gamma(y_t - L_t) + (1-\gamma)S_{t-M}$

Modelos de Regressão – Modelos de Suavização – checkresiduals

- O R possui uma função chamada checkresiduals() que computa os valores de ACF e testes estatísticos de forma automática.



Modelos de Regressão – Modelos de Suavização

- Comparando

Modelo de Suavização Exponencial com Tendência e Sazonalidade Aditiva (AAA)					
	ME	RMSE	MAE	MPE	MAPE
Training set	4,6402	56,6468	45,1723	0,1834	2,5685
Test set	82,3928	118,6700	98,3372	4,0307	4,9055

Modelo de Suavização Exponencial com Tendência Multiplicativa (MMN)					
	ME	RMSE	MAE	MPE	MAPE
Training set	- 0,6596	151,3118	126,9160	- 0,7679	7,3756
Test set	88,1354	166,4706	150,8469	3,9163	7,4725

Modelo de Suavização sem tendência e sazonalidade					
	ME	RMSE	MAE	MPE	MAPE
Training set	4,8306	151,3034	127,9211	- 0,4566	7,4106
Test set	109,0143	179,0154	160,6982	4,9657	7,9065

Modelo de Média Móvel MA(12)					
	ME	RMSE	MAE	MPE	MAPE
Training set	4,7570	143,8445	117,0441	- 0,4299	6,8130
Test set	54,7296	152,1767	135,1550	2,2278	6,7771

Modelo Com Sazonalidade e Tendência Quadrática					
	ME	RMSE	MAE	MPE	MAPE
Training set	0,0000	66,7614	51,9509	- 0,1526	3,0155
Test set	- 126,1654	153,2507	131,7250	- 6,4315	6,6987

Modelo Sazonal					
	ME	RMSE	MAE	MPE	MAPE
Training set	0,0000	96,3404	75,1310	- 0,3099	4,3269
Test set	217,9267	229,6509	217,9267	10,8646	10,8646

Modelo de Tendência Exponencial					
	ME	RMSE	MAE	MPE	MAPE
Training set	7,3750	159,0912	130,5631	- 0,4346	7,5540
Test set	203,2733	247,7550	216,2353	9,7213	10,4680

Modelo de Tendência Quadrático					
	ME	RMSE	MAE	MPE	MAPE
Training set	0,0000	146,9785	120,3980	- 0,7350	7,0158
Test set	- 83,9621	179,8494	133,7383	- 4,7254	7,0757

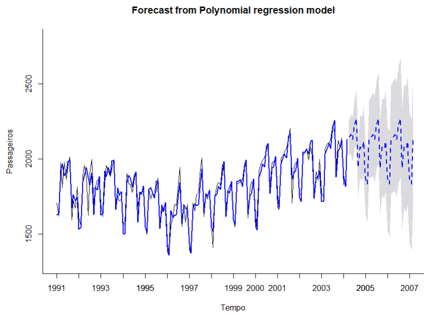
Modelo de Tendência Linear					
	ME	RMSE	MAE	MPE	MAPE
Training set	- 0,0000	158,9269	129,6778	- 0,8540	7,5360
Test set	193,1316	239,4863	209,4371	9,2099	10,1477

Modelo de Tendência Naive					
	ME	RMSE	MAE	MPE	MAPE
Training set	2,4509	168,1470	125,2975	- 0,3460	7,2714
Test set	- 14,7177	142,7551	115,9234	- 1,2750	6,0214

Modelos de Regressão – Modelos de Suavização

- Projetar para os próximos 36 meses no futuro

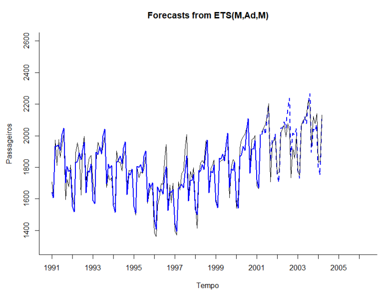
```
#primeiramente reestimamos o modelo com todos os
dados de treinamento e validacao
modelo_ses_final <- ets(passageiros_ts, model = "AAA")

#sumario do modelo
summary(modelo_ses_final)

#projeta os próximos 36 meses do futuro
modelo_ses_final_proj <- forecast(modelo_ses_final,
h=36, level=0.95)

#plota o grafico da serie temporal de treinamento e teste
plot(modelo_ses_final_proj, xlab="Tempo",
ylab="Passageiros", ylim=c(1300, 2800), xlim=c(1991,
2007), bty="l", flty=2, main="Forecast from Polynomial
regression model")
axis(1, at=seq(1991, 2007, 1), labels=format(seq(1991,
2007,1)))
lines(modelo_ses_final_proj$fitted, lwd=2, col="blue")
```



Modelos de Suavização – Suavização Exponencial Modelos Complexos: Série com tendência aditiva e sazonalidade multiplicativa (Holt-Winter)

- Utilizando o mesmo exemplo dos exercícios anteriores vamos estimar o modelo exp para o período de treinamento e projetar para o período de validação.

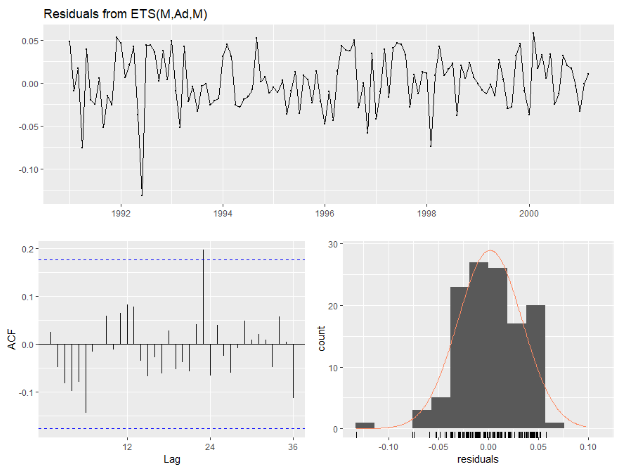
```
#estima o modelo de suavizacao na base de treinamento
modelo_ses <- ets(treinamento_ts, model = "MAM")

#resumo modelo
summary(modelo_ses)

#projeta os proximos 12 meses
modelo_ses_proj <- forecast(modelo_ses, h=tam_amostra_teste,
level=0.95)


ETS(M,Ad,M)

Call:
 ets(y = treinamento_ts, model = "MAM")

  Smoothing parameters:
    alpha = 0.6039
    beta  = 0.0032
    gamma = 1e-04
    phi   = 0.98
```



- $F_{t+k} = (L_t + \varphi k T_t) S_{t+k-M}$
- $L_t = \frac{\alpha y_t}{S_{t-M}} + (1-\alpha)(L_{t-1} + \varphi T_{t-1})$
- $T_t = \beta(L_t - L_{t-1}) + (1-\beta)\varphi T_{t-1}$
- $S_t = \gamma\left(\frac{y_t}{L_t + \varphi T_t}\right) + (1-\gamma) S_{t-M}$

Modelos de Regressão – Modelos de Suavização – checkresiduals

- O R possui uma função chamada checkresiduals() que computa os valores de ACF e testes estatísticos de forma automática.



Modelos de Regressão – Modelos de Suavização

- Comparando

Modelo de Suavização Exponencial com Tendência Aditiva e Sazonalidade Multiplicativa (MAdM)	ME	RMSE	MAE	MPE	MAPE
Training set	2,7847	56,6987	45,0034	0,0737	2,5660
Test set	34,9104	80,9640	65,6807	1,6861	3,3288

Modelo de Suavização Exponencial com Tendência e Sazonalidade Aditiva (AAA)	ME	RMSE	MAE	MPE	MAPE
Training set	4,6402	56,6468	45,1723	0,1834	2,5685
Test set	82,3928	118,6700	98,3372	4,0307	4,9055

Modelo de Suavização Exponencial com Tendência Multiplicativa (MMN)	ME	RMSE	MAE	MPE	MAPE
Training set	- 0,6596	151,3118	126,9160	- 0,7679	7,3756
Test set	88,1354	166,4706	150,8469	3,9163	7,4725

Modelo de Suavização sem tendência e sazonalidade	ME	RMSE	MAE	MPE	MAPE
Training set	4,8306	151,3034	127,9211	- 0,4566	7,4106
Test set	109,0143	179,0154	160,6982	4,9657	7,9065

Modelo de Média Móvel MA(12)	ME	RMSE	MAE	MPE	MAPE
Training set	4,7570	143,8445	117,0441	- 0,4299	6,8130
Test set	54,7296	152,1767	135,1550	2,2278	6,7771

Modelo Com Sazonalidade e Tendência Quadrática	ME	RMSE	MAE	MPE	MAPE
Training set	0,0000	66,7614	51,9509	- 0,1526	3,0155
Test set	- 126,1654	153,2507	131,7250	- 6,4315	6,6987

Modelo Sazonal	ME	RMSE	MAE	MPE	MAPE
Training set	0,0000	96,3404	75,1310	- 0,3099	4,3269
Test set	217,9267	229,6509	217,9267	10,8646	10,8646

Modelo de Tendência Exponencial	ME	RMSE	MAE	MPE	MAPE
Training set	7,3750	159,0912	130,5631	- 0,4346	7,5540
Test set	203,2733	247,7550	216,2353	9,7213	10,4680

Modelo de Tendência Quadrático	ME	RMSE	MAE	MPE	MAPE
Training set	0,0000	146,9785	120,3980	- 0,7350	7,0158
Test set	- 83,9621	179,8494	133,7383	- 4,7254	7,0757

Modelo de Tendência Linear	ME	RMSE	MAE	MPE	MAPE
Training set	- 0,0000	158,9269	129,6778	- 0,8540	7,5360
Test set	193,1316	239,4863	209,4371	9,2099	10,1477

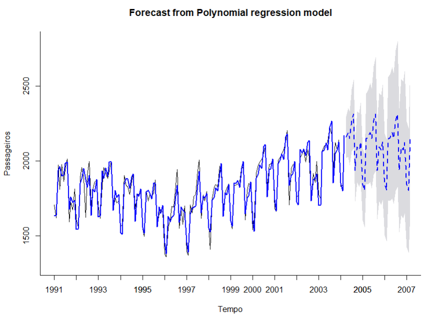
Modelo de Tendência Naive	ME	RMSE	MAE	MPE	MAPE
Training set	2,4509	168,1470	125,2975	- 0,3460	7,2714
Test set	- 14,7177	142,7551	115,9234	- 1,2750	6,0214

Modelos de Regressão – Modelos de Suavização

FGV

- Projetar para os próximos 36 meses no futuro

```
#primeiramente reestimamos o modelo com todos os
dados de treinamento e validacao
modelo_ses_final <- ets(passageiros_ts, model =
"MAM")

#sumario do modelo
summary(modelo_ses_final)

#projeta os próximos 36 meses do futuro
modelo_ses_final_proj <- forecast(modelo_ses_final,
h=36, level=0.95)

#plota o grafico da serie temporal de treinamento e teste
plot(modelo_ses_final_proj, xlab="Tempo",
ylab="Passageiros", ylim=c(1300, 2800), xlim=c(1991,
2007), bty="l", flty=2, main="Forecast from Polynomial
regression model")
axis(1, at=seq(1991, 2007, 1), labels=format(seq(1991,
2007,1)))
lines(modelo_ses_final_proj$fitted, lwd=2, col="blue")
```

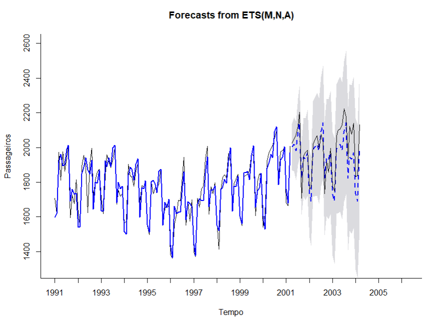


Modelos de Suavização – Suavização Exponencial Modelos Complexos: Usando o otimizador de modelos da função ETS – model(ZZZ)

FGV

- Utilizando o mesmo exemplo dos exercícios anteriores vamos estimar o modelo com o otimizador para período de treinamento e projetar para o período de validação.

```
#estima o modelo de suavizacao na base de treinamento
modelo_ses <- ets(treinamento_ts, model = "ZZZ", restrict = FALSE,
allow.multiplicative.trend = TRUE)

#resumo modelo
summary(modelo_ses)

#projeta os proximos 12 meses
modelo_ses_proj <- forecast(modelo_ses, h=tam_amostra_teste,
level=0.95)
```




```
ETS(M,N,A)

Call:
 ets(y = treinamento_ts, model = "ZZZ", restrict = FALSE,
allow.multiplicative.trend = TRUE)

  Smoothing parameters:
    alpha = 0.555
    gamma = 1e-04
```

Modelo de Suavização com Algoritmo de Otimização - MNA

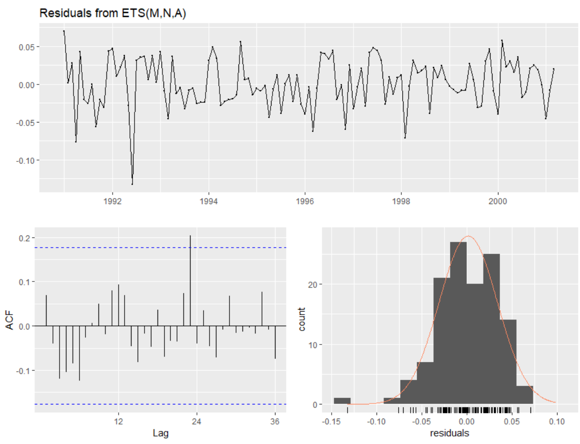
	ME	RMSE	MAE	MPE	MAPE
Training set	1,9857	56,8625	45,2133	0,0288	2,5770
Test set	49,7274	88,6281	73,9638	2,3912	3,7072

- O modelo selecionado pelo algoritmo são de erros multiplicativos, sem tendência e com sazonalidade multiplicativa. Apresenta um bom fit na amostra de treinamento, mas performa pior na amostra de teste (overfitting?).

- Note que a projeção já apresenta uma tendência e os valores não são mais iguais para todos os períodos.

Modelos de Regressão – Modelos de Suavização – checkresiduals

- O R possui uma função chamada checkresiduals() que computa os valores de ACF e testes estatísticos de forma automática.



Modelos de Regressão – Modelos de Suavização

- Comparando

Modelo de Suavização com Algoritmo de Otimização - MNA					
	ME	RMSE	MAE	MPE	MAPE
Training set	1,9857	56,8625	45,2133	0,0288	2,5770
Test set	49,7274	88,6281	73,9638	2,3912	3,7072

Modelo de Suavização Exponencial com Tendência Aditiva e Sazonalidade Multiplicativa (MAdM					
	ME	RMSE	MAE	MPE	MAPE
Training set	2,7847	56,6987	45,0034	0,0737	2,5660
Test set	34,9104	80,9640	65,6807	1,6861	3,3288

Modelo de Suavização Exponencial com Tendência e Sazonalidade Aditiva (AAA)					
	ME	RMSE	MAE	MPE	MAPE
Training set	4,6402	56,6468	45,1723	0,1834	2,5685
Test set	82,3928	118,6700	98,3372	4,0307	4,9055

Modelo de Suavização Exponencial com Tendência Multiplicativa (MMN)					
	ME	RMSE	MAE	MPE	MAPE
Training set	- 0,6596	151,3118	126,9160	- 0,7679	7,3756
Test set	88,1354	166,4706	150,8469	3,9163	7,4725

Modelo de Suavização sem tendência e sazonalidade					
	ME	RMSE	MAE	MPE	MAPE
Training set	4,8306	151,3034	127,9211	- 0,4566	7,4106
Test set	109,0143	179,0154	160,6982	4,9657	7,9065

Modelo de Média Móvel MA(12)					
	ME	RMSE	MAE	MPE	MAPE
Training set	4,7570	143,8445	117,0441	- 0,4299	6,8130
Test set	54,7296	152,1767	135,1550	2,2278	6,7771

Modelo Com Sazonalidade e Tendência Quadrática					
	ME	RMSE	MAE	MPE	MAPE
Training set	0,0000	66,7614	51,9509	- 0,1526	3,0155
Test set	- 126,1654	153,2507	131,7250	- 6,4315	6,6987

Modelo Sazonal					
	ME	RMSE	MAE	MPE	MAPE
Training set	0,0000	96,3404	75,1310	- 0,3099	4,3269
Test set	217,9267	229,6509	217,9267	10,8646	10,8646

Modelo de Tendência Exponencial					
	ME	RMSE	MAE	MPE	MAPE
Training set	7,3750	159,0912	130,5631	- 0,4346	7,5540
Test set	203,2733	247,7550	216,2353	9,7213	10,4680

Modelo de Tendência Quadrático					
	ME	RMSE	MAE	MPE	MAPE
Training set	0,0000	146,9785	120,3980	- 0,7350	7,0158
Test set	- 83,9621	179,8494	133,7383	- 4,7254	7,0757

Modelo de Tendência Linear					
	ME	RMSE	MAE	MPE	MAPE
Training set	- 0,0000	158,9269	129,6778	- 0,8540	7,5360
Test set	193,1316	239,4863	209,4371	9,2099	10,1477

Modelo de Tendência Naive					
	ME	RMSE	MAE	MPE	MAPE
Training set	2,4509	168,1470	125,2975	- 0,3460	7,2714
Test set	- 14,7177	142,7551	115,9234	- 1,2750	6,0214

Modelos de Regressão – Modelos de Suavização

FGV

- Modelos possíveis de suavização com a função ETS

	Seasonality		
Trend	None	Additive	Multiplicative
None	ZNN	ZNA	ZNM
Additive	ZAN	ZAA	ZAM
Multiplicative	ZMN	ZMA	ZMM
Additive damped	ZAdN	ZAdA	ZAdM
Multiplicative damped	ZMdN	ZMdA	ZMdM

Sumário

FGV

- Apresentações
- Regras do Jogo
- Introdução as Séries Temporais e Projeção
- Dados no contexto de Séries Temporais
- Avaliação de Performance
- Métodos de Projeção: Visão Geral
- Métodos de Regressão
- Métodos de Projeção: Suavização
- Modelos de Regressão ARIMA

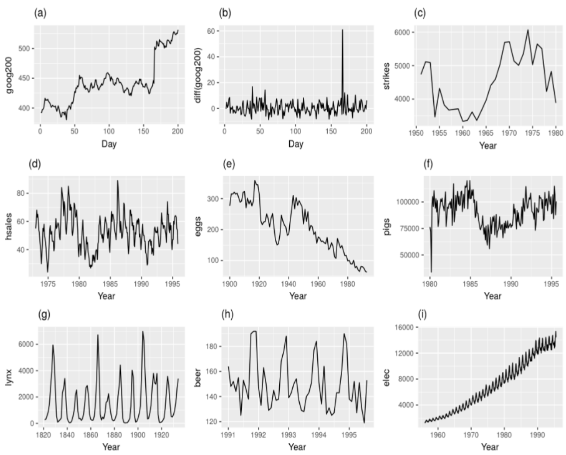
Modelos de Regressão – Modelos ARIMA

- Modelos ARIMA oferecem uma abordagem adicional para a projeção de séries temporais;
- Enquanto modelos de suavização exponencial são baseados na descrição de tendência e sazonalidade na série de dados, modelos ARIMA são utilizados para descrever/modelar a autocorrelação nos dados;
- Antes de introduzirmos modelos de ARIMA os conceitos de estacionariedade e diferenciação de séries temporais precisam ser analisados.

Modelos de Regressão – Modelos ARIMA: Estacionariedade

- Uma séries de dados estacionária é aquela que não depende do período onde a série é observada. Mais precisamente, se y_t é estacionária então a distribuição de $y_t, y_{t+1} + \cdots + y_{t+T}$ não depende de t.
- Séries temporais com tendência e/ou sazonalidade não são estacionárias, pois a tendência e sazonalidade irão afetar os valores da série temporal em tempos diferentes.
- A série de ruído branco (White noise) é estacionária.

Modelos de Regressão – Modelos ARIMA: Estacionariedade



Quais séries são estacionárias?

- Séries d, h e i são descartadas devido a sazonalidade;
- Tendência e troca de nível removem as séries a, c, e, f e i;
- A séries remanescentes b e g são estacionárias;
- A uma primeira vista existem ciclos em g que podem aparentar uma sazonalidade, porém os ciclos não são periódicos e repetidos em intervalos iguais que são característicos da sazonalidade. São oscilações aleatórias características de estacionariedade.

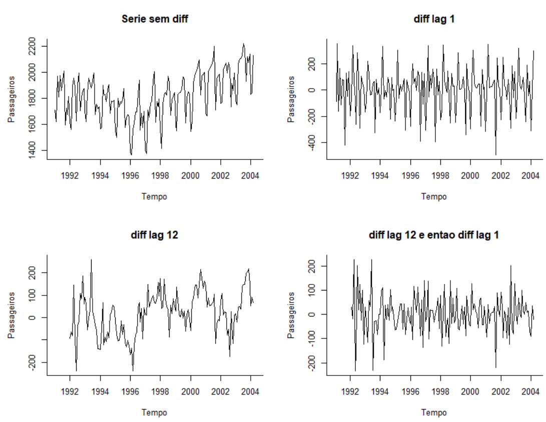
Modelos ARIMA: Diferenciação

- Um método simples para remoção de tendência e/ou sazonalidade é a chamada diferenciação y'_t;
- Diferenciação significa tirar a diferença entre dois valores. Tirar a diferença de um período (lag-1) significa tirar a diferença entre dois valores consecutivos na série temporal:

$$\Delta y_t = y_t - y_{t-1}$$

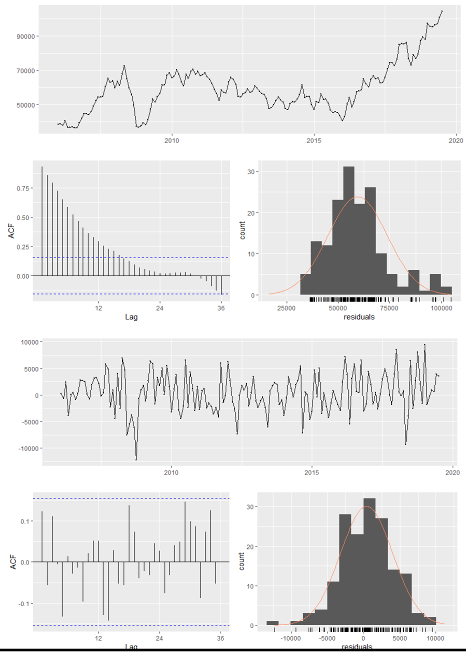
- O comando em R para diferenciação é a função diff.
- Lag-1 remove a tendência;
- Lag-m, m é a frequência da sazonalidade, remove a sazonalidade;
- Lag-m e depois Lag-1 remove tendência e sazonalidade.

- A diferenciação é comumente utilizada como uma técnica de pré-processamento dos dados (remoção de tendência e sazonalidade) antes de conduzir a estimação dos modelos.



Modelos ARIMA: Diferenciação - Ibov

```
#carrega o arquivo com a seria
temporal Ibov
Ibov <- read.csv("Ibovespa.csv")

#transforma em série temporal
Ibov_ts <- ts(Ibov$Adj.Close,
start=c(2006,1), frequency = 12)

#plot gráfico
par(mfrow=c(1,1))

plot(Ibov_ts)

#checa a autocorrelação
checkresiduals(Ibov_ts)
Box.test(Ibov_ts, type="Ljung")

#diferencia 1 vez
Ibov_ts_diff <- diff(Ibov_ts, lag=1)

#checa a autocorrelação
checkresiduals(Ibov_ts_diff)
Box.test(Ibov_ts_diff, type="Ljung")
```



Modelos ARIMA: Diferenciação: Testes de Raiz Unitária

- Uma maneira mais objetiva de determinar a necessidade de diferenciar uma série temporal é o teste de raiz unitária;
- Testes de Raiz Unitária são testes de hipóteses estatísticas para estacionariedade desenhados para determinar se diferenciação é necessária;
- Existem vários tipos de testes unitários, dos quais apresentaremos dois o KPSS e ADF.
- Em casos específicos os testes podem apresentar resultados contraditório. Portanto, recomenda-se a aplicação de ambos os testes para avaliar a estacionariedade das séries.

Modelos ARIMA: Diferenciação: Testes de Raiz Unitária

KPSS

- Neste teste a hipótese nula é que os dados são estacionários. Portanto, se a hipótese nula for refutada os dados não são estacionários.
- No R temos a função ur.kpss do pacote URCA que executa o teste de KPSS.
- Executando para a base bruta do Ibov temos o resultado ao lado. O valor da estatística de teste é 1.1483, um valor muito maior que o intervalo de confiança para 1%, ou seja, 0.739. Portanto, refutamos a hipótese de estacionariedade e consideramos a série IBOV não estacionária.

KPSS

```
#Carrega Biblioteca de testes estatísticos
library(urca)

#executa o teste de KPSS
summary(ur.kpss(Ibov_ts))

#######################
# KPSS Unit Root Test #
#######################

Test is of type: mu with 4 lags.

Value of test-statistic is: 1.1483

Critical value for a significance level of:
                10pct  5pct 2.5pct  1pct
critical values 0.347 0.463  0.574 0.739
```

Modelos ARIMA: Diferenciação: Testes de Raiz Unitária

KPSS

- Executando para a base diferenciada do Ibov temos o resultado ao lado. O valor da estatística de teste é 0.1603, um valor menor que o intervalo de confiança para 1%, ou seja, 0.739. Portanto, não refutamos a hipótese de estacionariedade e consideramos a série IBOV estacionária.

KPSS

```
#Carrega Biblioteca de testes estatísticos
library(urca)

#executa o teste de KPSS
summary(ur.kpss(Ibov_ts_diff))

#######################
# KPSS Unit Root Test #
#######################

Test is of type: mu with 4 lags.

Value of test-statistic is: 0.1603

Critical value for a significance level of:
                10pct  5pct 2.5pct  1pct
critical values 0.347 0.463  0.574 0.739
```

Modelos ARIMA: Diferenciação: Testes de Raiz Unitária

ADF

- Neste teste a hipótese nula é que os dados **não são** estacionários. Portanto, se a hipótese nula for refutada os dados são estacionários.
- No R temos a função ur.df do pacote URCA que executa o teste de ADF.
- Executando para a base bruta do Ibov temos o resultado ao lado. O valor da estatística de teste é 1.0668, um valor muito maior que o intervalo de confiança para 1%, ou seja, -2.58. Portanto, não refutamos a hipótese de não estacionariedade e consideramos a série IBOV não estacionária.

ADF

```
#Carrega Biblioteca de testes estatísticos
library(urca)

#executa o teste de ADF
summary(ur.df(Ibov_ts))

Value of test-statistic is: 1.0668

Critical values for test statistics:
      1pct  5pct 10pct
tau1 -2.58 -1.95 -1.62
```

Modelos ARIMA: Diferenciação: Testes de Raiz Unitária

ADF

- Executando para a base bruta do Ibov temos o resultado ao lado. O valor da estatística de teste é -8.746, um valor menor que o intervalo de confiança para 1%, ou seja, -2.58. Portanto, refutamos a hipótese de não estacionariedade e consideramos a série IBOV estacionária.

ADF

```
#Carrega Biblioteca de testes estatísticos
library(urca)

#executa o teste de ADF
summary(ur.df(Ibov_ts_diff))

Value of test-statistic is: -8.746

Critical values for test statistics:
      1pct  5pct 10pct
tau1 -2.58 -1.95 -1.62
```

Modelos ARIMA: Modelos de Autorregressão – AR(p)

- Quando utilizamos modelos de regressão linear para a projeção de séries temporais, temos a capacidade de modelar padrões como tendência e sazonalidade;
- Entretanto, modelos comuns de regressão não consideram a correlação entre valores em diferentes períodos do tempo;
- Em séries temporais, valores temporalmente próximos tendem a ser correlacionados. Por exemplo, os preços de ações em dias consecutivos após o anúncio de um fato relevante para a empresa;
- Este tipo de correlação dos dados é chamada de autocorrelação e pode possuir grande valor informativo, melhorando a qualidade das projeções.
- Para modelar a autocorrelação utilizamos modelos autorregressivos AR(p), de ordem p. Por exemplo, um modelo de AR(2) tem a seguinte forma:

$$y_t = \beta_0 + \beta_1 y_{t-1} + \beta_2 y_{t-2} + e_t$$

Modelos ARIMA: Modelos de Média Móvel – MA(q)

- Ao invés de utilizar valores passados da variável projetada, uma média móvel usa os valores de erros das projeções passadas, por exemplo para um modelo MA(2) temos:

$$y_t = \beta_0 + \beta_1 e_{t-1} + \beta_2 e_{t-2} + e_t$$

- Onde e_t é ruído branco;
- Este modelo considera que y_t pode ser modelado como uma média ponderada dos erros de projeção passados;
- Estes modelos não devem ser confundidos com os modelos de média móvel de suavização discutidos anteriormente. Um modelo MA é utilizado para projetar futuros valores, enquanto médias móveis são utilizadas para estimar os efeitos de tendência de períodos passados.

Modelos ARIMA: ARIMA(p,d,q)

- Combinando modelos de diferenciação (d), autorregressivos - AR(p), e de média móvel – MA(q) obtemos os modelos ARIMA. Ou seja, modelos autorregressivos integrados e média móvel. O modelo completo é escrito da seguinte forma:

$$\Delta y_t = \beta_0 + \beta_1 \Delta y_{t-1} + \cdots + \beta_p \Delta y_{t-p} + \theta_1 e_{t-1} + \cdots + \theta_q e_{t-q} + e_t$$

- Por exemplo, um modelo diferenciado uma vez (d=1), autorregressivo de ordem 2 - AR(p=2), e média móvel de ordem 1 – MA(q=1) é uma ARIMA(2,1,1) apresenta a seguinte forma:

$$\Delta y_t = \beta_0 + \beta_1 \Delta y_{t-1} + \beta_2 \Delta y_{t-2} + \theta_1 e_{t-1} + e_t$$

- Onde $\Delta y_t = y_t$- y_{t-1}

- Portando, para estimarmos um modelo ARIMA é necessário a definição dos parâmetros de diferenciação (d), autorregressão (p) e média móvel (q).

Modelos ARIMA: ARIMA(p,d,q) – Processo para estimação de modelos

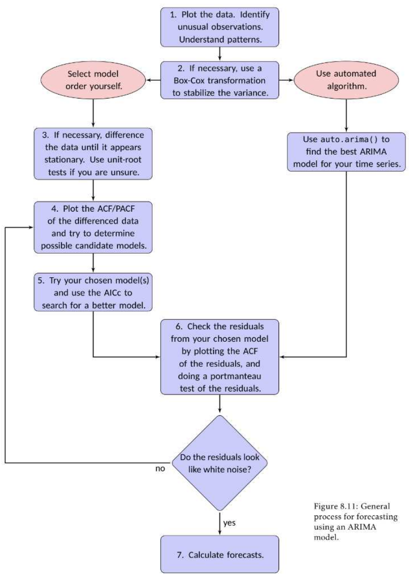


Figure 8.11: General process for forecasting using an ARIMA model.

Modelos ARIMA: ARIMA(p,d,q) – Definindo os parâmetros p, d e q

FGV

- Conforme visto anteriormente o nível de diferenciação d pode ser definido com base nos testes de KPSS e ADF. Realizando sucessivas diferenciações até os dados ficarem estacionários.

- Para os valores de p e q utilizaremos as análises de autocorrelação (ACF) autocorrelação parcial (PCF).

- Relembrando que a ACF apresenta as correlações entre y_t e y_{t-1} até y_{t-k} para diferentes valores de k. Agora se y_t e y_{t-1} são correlacionados então y_{t-1} e y_{t-2} também são correlacionados. Entretanto, y_t e y_{t-2} podem ser correlacionados simplesmente pelo fato de ambos serem correlacionados com y_{t-1} e não pelo fato de existir informação em y_{t-2} que ajuda a prever y_t. Para resolver este problema utilizamos a função de autocorrelação parcial (PCF).

- A função PCF calcula a autocorrelação entre y_t e y_{t-k} após remover os efeitos de y_{t-1} até y_{t-k-1}, ou seja, os efeitos dos lags intermediários. A interpretação dos resultados é a mesma do ACF, analisando um intervalo de confiança de 95%.

Modelos ARIMA: ARIMA(p,d,q) – Definindo os parâmetros p, d e q

FGV

- Em R podemos usar as funções Acf e Pacf para calcular os valores de Autocorrelação e Autocorrelação Parcial.

#Passageiros diferenciado sazonalmente 12
treinamento_ts_diff <- diff(treinamento_ts, lag=12)

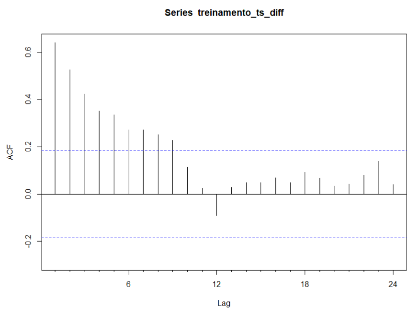
ajuste_sazonal <- treinamento_ts-treinamento_ts_diff

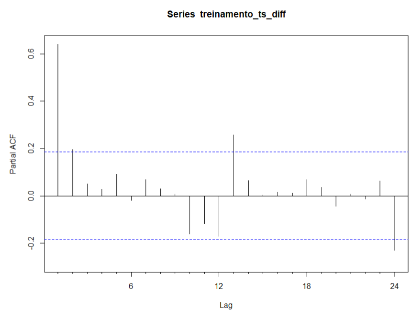
#executa o teste de ADF
summary(ur.kpss(treinamento_ts_diff))

#executa o teste de ADF
summary(ur.df(treinamento_ts_diff))

#calcula a ACF
Acf(treinamento_ts_diff)

#calcula a PCF
Pacf(treinamento_ts_diff)





Modelos ARIMA: ARIMA(p,d,q) – Definindo os parâmetros p, d e q

FGV

Podemos usar os resultados da ACF e PCF para parametrizar o modelo ARIMA da seguinte maneira:

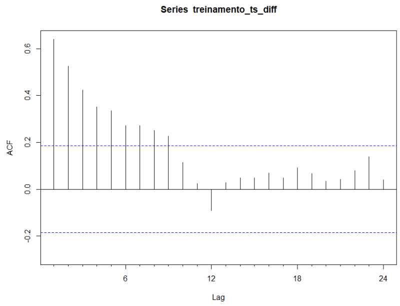
ARIMA(p,d,0):

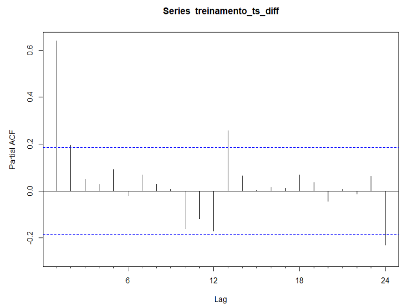
- A ACF apresenta decaimento exponencial ou senoidal (alternando entre positivo e negativo);
- Existe um pico no lag p na PCF, mas nada mais após o lag p.

ARIMA(0,d,q):

- A PCF apresenta decaimento exponencial ou senoidal (alternando entre positivo e negativo);
- Existe um pico no lag q na ACF, mas nada mais após o lag q.

No caso dos passageiros podemos testar um modelo ARIMA(2,1,1).



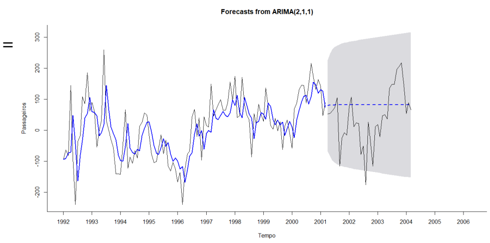


Modelos ARIMA: ARIMA(p,d,q) – Definindo os parâmetros p, d e q

FGV

Utilizando a função Arima para estimar um modelo ARIMA(p,d,q) no R.

```
#Modelo Arima
Modelo_ARIMA <- Arima(treinamento_ts_diff, order = c(2,1,1))

#resumo modelo
summary(Modelo_ARIMA)

#projeta os proximos 12 meses
modelo_ARIMA_proj <- forecast(Modelo_ARIMA, h=tam_amostra_teste, level=0.95)

> summary(Modelo_ARIMA)
Series: treinamento_ts_diff
ARIMA(2,1,1)

Coefficients:
        ar1     ar2      ma1
     0.4664  0.1530  -0.9413
s.e. 0.1123  0.1069   0.0595

sigma^2 estimated as 5636:  log likelihood=-630.03
AIC=1268.06   AICc=1268.44   BIC=1278.86
```



$$\Delta y_t = \beta_0 + \beta_1 \Delta y_{t-1} + \beta_2 \Delta y_{t-2} + \theta_1 e_{t-1} + e_t$$

$$\Delta y_t = 0.0 + 0.4664 \Delta y_{t-1} + 0.1530 \Delta y_{t-2} - 0.9413 e_{t-1}$$

Modelos ARIMA: ARIMA(p,d,q) – Auto Arima

Em R temos a função que estima automaticamente todos os parâmetros do ARIMA. A função auto.arima.

```
> #função auto.arima
> auto.arima(treinamento_ts_diff, seasonal = FALSE, stepwise=FALSE, approximation = FALSE)
Series: treinamento_ts_diff
ARIMA(1,1,1)

Coefficients:
         ar1      ma1
      0.3667  -0.8109
s.e.  0.2030   0.1516

sigma^2 estimated as 5680:  log likelihood=-630.79
AIC=1267.58   AICc=1267.81   BIC=1275.68
```

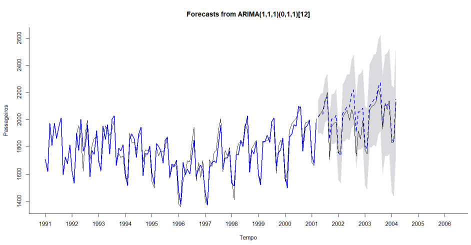
Neste caso o R sugere um modelo ARIMA(1,1,1) que é mais simples que o anterior e possui uma medida AIC mais baixa, indicando um fit melhor nos dados de treinamento.

Entretanto, podemos notar que o fato de não considerar a sazonalidade tornar o modelo ARIMA limitado.

Modelos S-ARIMA: Modelos ARIMA com sazonalidade

Utilizando a função auto.arima vamos estimar um modelo SARIMA.

```
#Modelo Arima
Modelo_ARIMA <- auto.arima(treinamento_ts,
stepwise=FALSE, approximation = FALSE)


#resumo modelo
summary(Modelo_ARIMA)

#projeta os proximos 12 meses
modelo_ARIMA_proj <- forecast(Modelo_ARIMA,
h=tam_amostra_teste, level=0.95)

> summary(Modelo_ARIMA)
Series: treinamento_ts
ARIMA(1,1,1)(0,1,1)[12]

Coefficients:
         ar1      ma1     sma1
      0.3221  -0.7446  -0.7017
s.e.  0.1644   0.1185   0.1021

sigma^2 estimated as 4032:  log likelihood=-615.44
AIC=1238.88   AICc=1239.26   BIC=1249.68
```

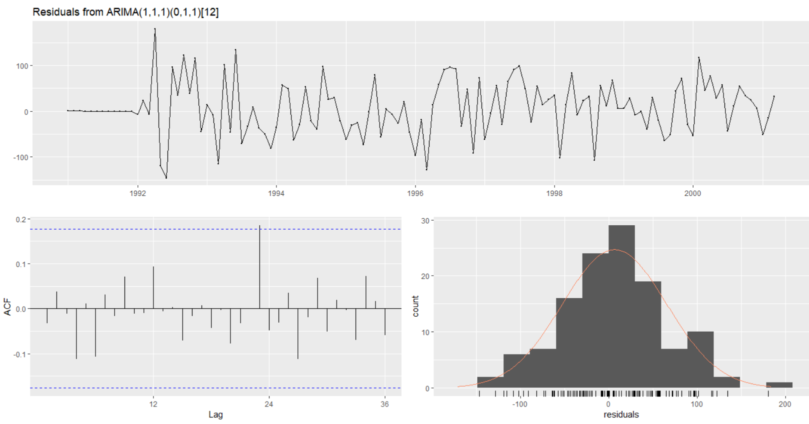


- O modelo estimado pelo auto.arima foi um modelo SARIMA(1,1,1)(0,1,1), ou seja, AR(1), diferenciado 1 e MA(1) na parte sem sazonalidade e AR(0), diferenciado 1 e MA(0) na sazonalidade.

Modelo de SARIMA					
	ME	RMSE	MAE	MPE	MAPE
Training set	6,7546	59,2256	45,5480	0,2971	2,5933
Test set	- 45,9860	76,5004	53,0434	- 2,3883	2,7392

Modelos S-ARIMA: Modelos ARIMA com sazonalidade

- O R possui uma função chamada checkresiduals() que computa os valores de ACF e testes estatísticos de forma automática.



Modelos S-ARIMA: Modelos ARIMA com sazonalidade

- Projetar para os próximos 36 meses no futuro

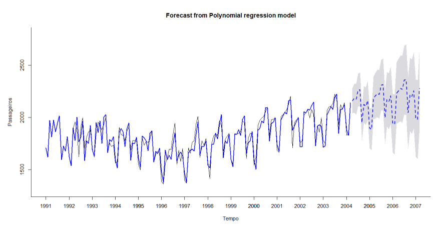
#Preparar projecao

#primeiramente reestimamos o modelo com todos os dados de treinamento e validacao
Modelo_ARIMA_final <- auto.arima(passageiros_ts, stepwise=FALSE, approximation = FALSE)

#sumario do modelo
summary(Modelo_ARIMA_final)

#projeta os próximos 36 meses do futuro
Modelo_ARIMA_final_proj <- forecast(Modelo_ARIMA_final, h=36, level=0.95)

#plota o grafico da serie temporal de treinamento e teste
plot(Modelo_ARIMA_final_proj, xlab="Tempo", ylab="Passageiros", ylim=c(1300, 2800), xlim=c(1991, 2007), bty="l", flty=2, main="Forecast from Polynomial regression model")
axis(1, at=seq(1991, 2007, 1), labels=format(seq(1991, 2007,1)))
lines(Modelo_ARIMA_final_proj$fitted, lwd=2, col="blue")



Resultado Final

FGV

- Comparando

Modelo de SARIMA					
	ME	RMSE	MAE	MPE	MAPE
Training set	6,7546	59,2256	45,5480	0,2971	2,5933
Test set	- 45,9860	76,5004	53,0434	- 2,3883	2,7392

Modelo de Suavização com Algoritmo de Otimização - MNA					
	ME	RMSE	MAE	MPE	MAPE
Training set	1,9857	56,8625	45,2133	0,0288	2,5770
Test set	49,7274	88,6281	73,9638	2,3912	3,7072

Modelo de Suavização Exponencial com Tendência Aditiva e Sazonalidade Multiplicativa (MAdM)					
	ME	RMSE	MAE	MPE	MAPE
Training set	2,7847	56,6987	45,0034	0,0737	2,5660
Test set	34,9104	80,9640	65,6807	1,6861	3,3288

Modelo de Suavização Exponencial com Tendência e Sazonalidade Aditiva (AAA)					
	ME	RMSE	MAE	MPE	MAPE
Training set	4,6402	56,6468	45,1723	0,1834	2,5685
Test set	82,3928	118,6700	98,3372	4,0307	4,9055

Modelo de Suavização Exponencial com Tendência Multiplicativa (MMN)					
	ME	RMSE	MAE	MPE	MAPE
Training set	- 0,6596	151,3118	126,9160	- 0,7679	7,3756
Test set	88,1354	166,4706	150,8469	3,9163	7,4725

Modelo de Suavização sem tendência e sazonalidade					
	ME	RMSE	MAE	MPE	MAPE
Training set	4,8306	151,3034	127,9211	- 0,4566	7,4106
Test set	109,0143	179,0154	160,6982	4,9657	7,9065

Modelo de Média Móvel MA(12)					
	ME	RMSE	MAE	MPE	MAPE
Training set	4,7570	143,8445	117,0441	- 0,4299	6,8130
Test set	54,7296	152,1767	135,1550	2,2278	6,7771

Modelo Com Sazonalidade e Tendência Quadrática					
	ME	RMSE	MAE	MPE	MAPE
Training set	0,0000	66,7614	51,9509	- 0,1526	3,0155
Test set	- 126,1654	153,2507	131,7250	- 6,4315	6,6987

Modelo Sazonal					
	ME	RMSE	MAE	MPE	MAPE
Training set	0,0000	96,3404	75,1310	- 0,3099	4,3269
Test set	217,9267	229,6509	217,9267	10,8646	10,8646

Modelo de Tendência Exponencial					
	ME	RMSE	MAE	MPE	MAPE
Training set	7,3750	159,0912	130,5631	- 0,4346	7,5540
Test set	203,2733	247,7550	216,2353	9,7213	10,4680

Modelo de Tendência Quadrático					
	ME	RMSE	MAE	MPE	MAPE
Training set	0,0000	146,9785	120,3980	- 0,7350	7,0158
Test set	- 83,9621	179,8494	133,7383	- 4,7254	7,0757

Modelo de Tendência Linear					
	ME	RMSE	MAE	MPE	MAPE
Training set	- 0,0000	158,9269	129,6778	- 0,8540	7,5360
Test set	193,1316	239,4863	209,4371	9,2099	10,1477

Modelo de Tendência Naive					
	ME	RMSE	MAE	MPE	MAPE
Training set	2,4509	168,1470	125,2975	- 0,3460	7,2714
Test set	- 14,7177	142,7551	115,9234	- 1,2750	6,0214

FGV

OBRIGADO!