Национальный исследовательский университет

"ВЫСШАЯ ШКОЛА ЭКОНОМИКИ"

Факультет экономических наук

Курсовая работа

Нестатистические методы прогнозирования временных рядов

Выполнила
студентка группы 212И
Кузина А.В.

Научный руководитель
Старший преподаватель
департамента прикладной экономики
Демешев Б.Б.

Москва 2015 г.

Содержание

1 Введение

Данная работа представляет собой исследование и сравнение нескольких основных нестатистических методов прогнозирования временных рядов.

Под прогнозом понимают предсказание будущих значений ряда при помощи каких-либо научных методов. Существует множество ситуаций, в которых необходимо применение прогнозирования рядов. Так, любое эффективное планирование невозможно без прогноза одного или нескольких показателей.

При прогнозировании данных можно использовать только значения используемого ряда, а можно учитывать влияние других факторов, представляемых также в виде временных рядов. Нестатистические методы прогнозирования, рассматриваемые в данной работе, подразумевают первый вариант постановки задачи. Однако, в различных методах может учитываться наличие тренда или сезонных колебаний данных.

Целью данной работы является подробное описание нескольких нестатистических методов прогнозирования рядов и сравнение их прогнозной силы на различных видах данных. А также сделать вывод о том, всегда ли более сложные методы прогнозирования дают более точный прогноз и зависит ли результативность от метода, с помощью которого оценивается точность прогноза.

2 Методы прогнозирования

Введем общие обозначения для всех представленных ниже методов:

F_{t+h} - прогноз, сделанный в момент времени t на h периодов вперед,

а h назовем горизонтом прогноза;

Y_t - реальное значение ряда в период t

2.1 Naive

В даном случае прогноз на любое количество периодов вперед равен последнему наблюдению из данного ряда:

$$F_{t+h} = Y_t \tag{1}$$

2.2 Метод среднего арифметического

Прогноз также не зависит от количества периодов и равен среднему арифметическому всех используемых для прогноза наблюдений:

$$F_{t+h} = \frac{1}{t}\sum_{i=1}^{t} Y_t \tag{2}$$

2.3 Простое экспоненциальное сглаживание.

Этот метод в основном используется для прогнозирования рядов, которые не имеют очевидного тренда или особой модели поведения. Прогноз осуществляется при помощи предыдущих наблюдений взятых с весом, убывающим в геометрической прогресси по мере устаревания данных:

$$F_{t+1} = \alpha Y_t + \alpha(1-\alpha)Y_{t-1} + \alpha(1-\alpha)^2 Y_{t-2} ... = \alpha\sum_{i=0}^{t}(1-\alpha)^i Y_{t-i} \tag{3}$$

где $0 < \alpha \leq 1$ - параметр сглаживания.

Чем ближе значение параметра α к 0, тем большее значения придается более ранним данным. Соответственно, чем ближе значение α к единице, тем больший вес имеют новые данные.

Для удобства использования можно выразить прогноз на период $t+1$ через реальные данные и прогноз за предыдущий период t:

$$F_{t+1} = \alpha Y_t + (1-\alpha)F_t \tag{4}$$

Заметим также, что прогноз методом экспоненциального сглаживания не зависит от количества периодов, на который делается прогноз. То есть $F_{t+h} = F_{t+1}$

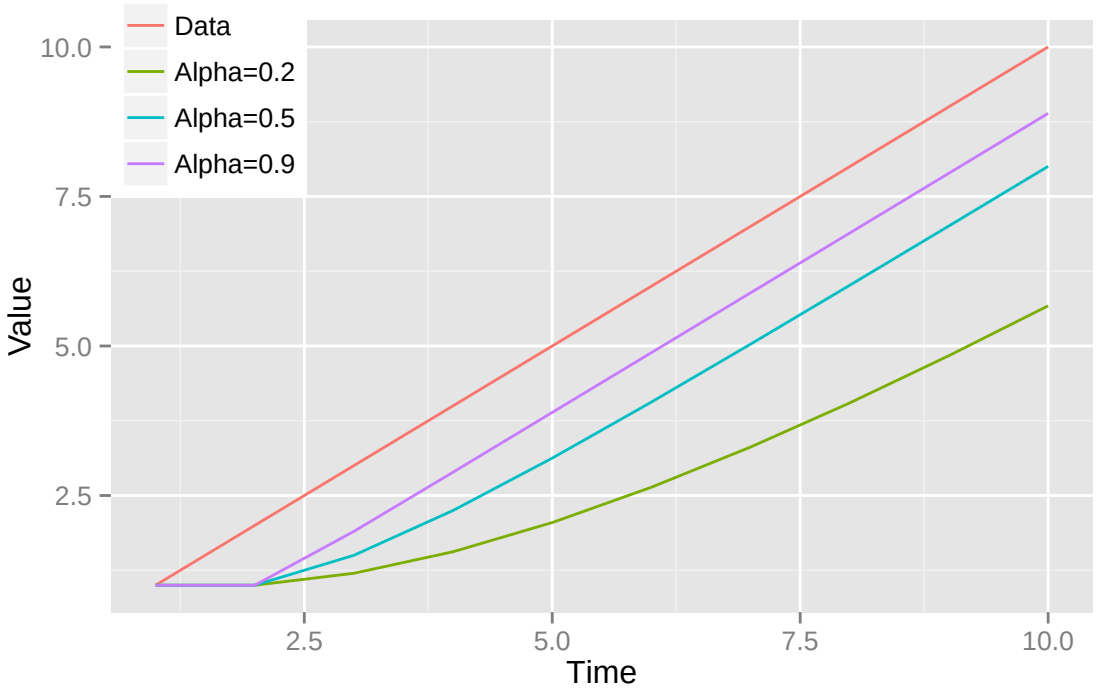
Пример 1.

Посмотрим, как работает экспоненциальное сглаживание на простых временных рядах. Для начала возьмем линейно возрастающий от 1 до 10 ряд и применим к нему функцию ses() при 3 различных значениях α. А затем, изобразим полученный результат на графике.

```
y1<- ses(ts(c(1:10)), alpha=0.2, initial="simple")
y2<- ses(ts(c(1:10)), alpha=0.5, initial="simple")
y3<- ses(ts(c(1:10)), alpha=0.9, initial="simple")

df<- data.frame(x=1:10, y=c(1:10),
                y1=y1$fitted, y2=y2$fitted, y3=y3$fitted)
df_melted <- melt(df, value.name ="y",
                measure.vars=c("y","y1","y2","y3") )
```

```
qplot(data=df_melted, x=x, y=y, color=variable, geom="line")+
  labs(list(title = "Figure 1. Simple exponential smoothing",
            x = "Time", y = "Value"))+
  scale_colour_discrete(
    name="",
    breaks=c("y", "y1", "y2","y3"),
    labels=c("Data", "Alpha=0.2", "Alpha=0.5","Alpha=0.9"))+
   theme(plot.title = element_text(size = 10, vjust = 2),
          legend.position = c(0.1, 0.9))
```

Figure 1. Simple exponential smoothing



Пример 2.

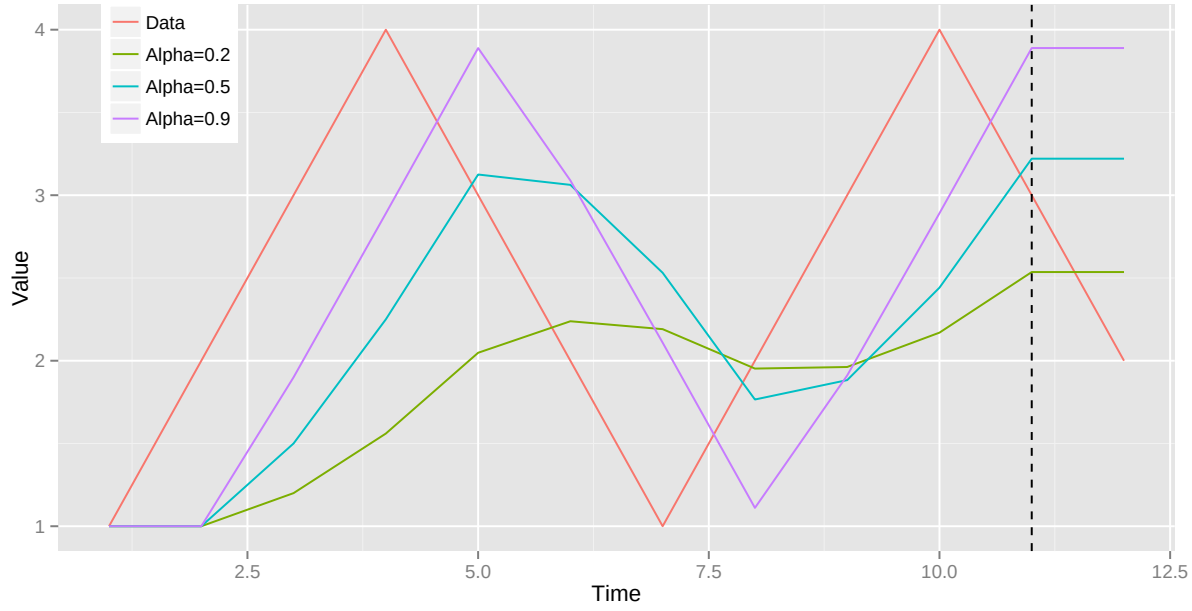
Аналогично, применим простое экспоненциальное сглаживание к ряду с изменяющимся трендом. Для этого сгенерируем вектор (1,2,3,4,3,2,1,2,3,4) и также воспользуемся функцией ses().

```
v <- c(1,2,3,4,3,2,1,2,3,4)
y1<- ses(ts(v), alpha=0.2, initial="simple", h=2)
y2<- ses(ts(v), alpha=0.5, initial="simple", h=2)
y3<- ses(ts(v), alpha=0.9, initial="simple", h=2)

df<- data.frame(x=1:12, y=c(v,3,2), y1=c(y1$fitted,y1$mean),
                y2=c(y2$fitted,y2$mean), y3=c(y3$fitted,y3$mean))
df_melted <- melt(df, value.name ="y",
                  measure.vars=c("y","y1","y2","y3") )
```

```
qplot(data=df_melted, x=x, y=y, color=variable, geom="line")+
  labs(list(
    title = "Figure 2. Simple exponential smoothing",
    x = "Time", y = "Value"))+
```

```
  geom_vline(xintercept=11, linetype="dashed",color="black")+
    scale_colour_discrete(
    name="",
    breaks=c("y", "y1", "y2","y3"),
    labels=c("Data", "Alpha=0.2", "Alpha=0.5","Alpha=0.9"))+
  theme(plot.title = element_text(size = 12, vjust = 2),
          legend.position = c(0.1, 0.9))
```

Figure 2. Simple exponential smoothing



Как видно на графиках обоих примеров, сглаженный ряд тем ближе к исходным данным, чем больше значение параметра. Однако прогноз на 11 и 12 период второго примера имеет достаточно низкую точность, так как в модели простого экспоненциального сглаживания не учитывается ни тренд, ни сезонность.

2.4 Метод Хольта

Хольт расширил метод простого экспоненциального сглаживания, разработав модель, которая позволяет делать более точный прогноз для данных с линейным трендом. Модель состоит из основного уравнения прогноза и двух сглаживающих уравнений, которые отвечают за уровень и тренд. Именно поэтому данную модель также называют двойным экспоненциальным сглаживанием.

$$F_{t+h} = l_t + hb_t \tag{5}$$

$$l_t = \alpha Y_t + (1-\alpha)(l_{t-1} + b_{t-1}) \tag{6}$$

$$b_t = \beta(l_t - l_{t-1}) + (1-\beta)b_{t-1} \tag{7}$$

Первое уравнение (F_{t+h}) является основным в данной модели. Оно показывает сам прогноз значения на период t+h, которое можно сделать на онове данных, имеющихся в момент t; l_t отвечает за уровень, а b_t - за тренд.

Заметим, что если убрать уравнение (7) из модели, то мы получим обычное экспоненциальное сглаживание. Учитывание тренда позволяет делать более точный прогноз на несколько периодов вперед. Однако, если в данных не наблюдается линейного тренда, эффективность использования этой модели для прогнозирования значительно снижается.

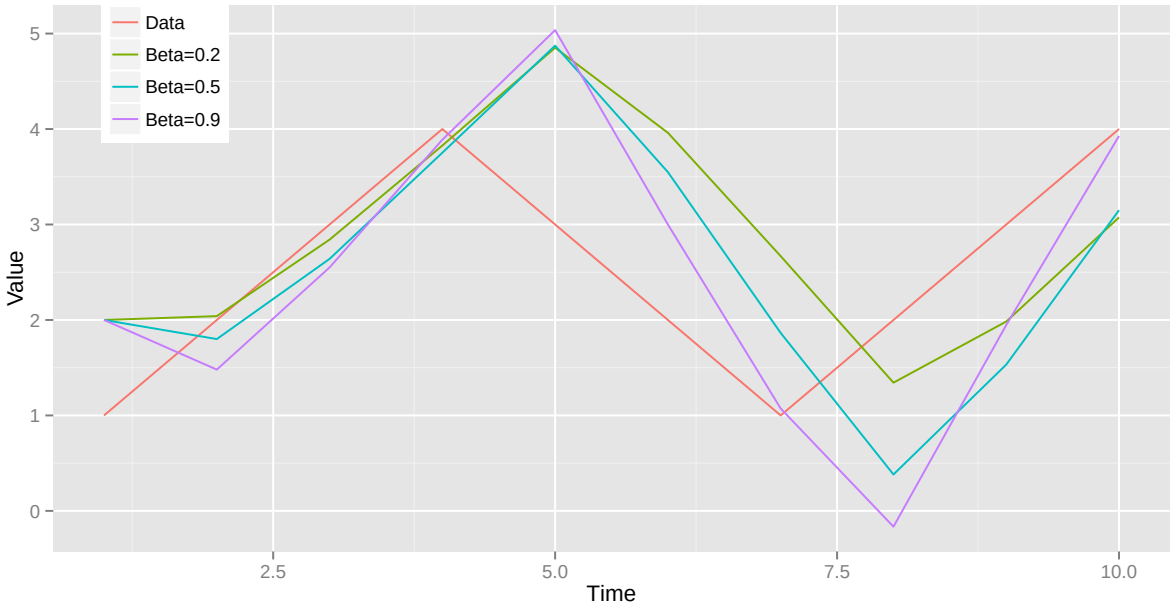
Пирмер 3.

Если взять линейный ряд, как в примере с экспоненциальным сглаживанием, то сглаженный по методу Хольта ряд полностью совпадет с исходным. Поэтому воспользуемся рядом из второго примера.

```
v <- c(1,2,3,4,3,2,1,2,3,4)
y1<- holt(ts(v), alpha=0.8, beta = 0.2, initial="simple")
y2<- holt(ts(v), alpha=0.8, beta = 0.5, initial="simple")
y3<- holt(ts(v), alpha=0.8, beta = 0.9, initial="simple")

df<- data.frame(x=1:10, y=v,
                y1=y1$fitted, y2=y2$fitted, y3=y3$fitted)
df_melted <- melt(df, value.name ="y",
                  measure.vars=c("y","y1","y2","y3") )
```

```
qplot(data=df_melted, x=x, y=y, color=variable, geom="line")+
  labs(list(
    title = "Figure 3. Holt's method",
    x = "Time", y = "Value"))+
  scale_colour_discrete(
    name="",
    breaks=c("y", "y1", "y2","y3"),
    labels=c("Data", "Beta=0.2", "Beta=0.5","Beta=0.9"))+
   theme(plot.title = element_text(size = 12, vjust = 2),
         legend.position = c(0.1, 0.9))
```

Figure 3. Holt's method



Можно заметить, что чем меньше значение параметра β, тем плавнее подстраивается сглаженный ряд под изменение тренда. Тем не менее, в данном случае точность прогноза значительно повысилась.

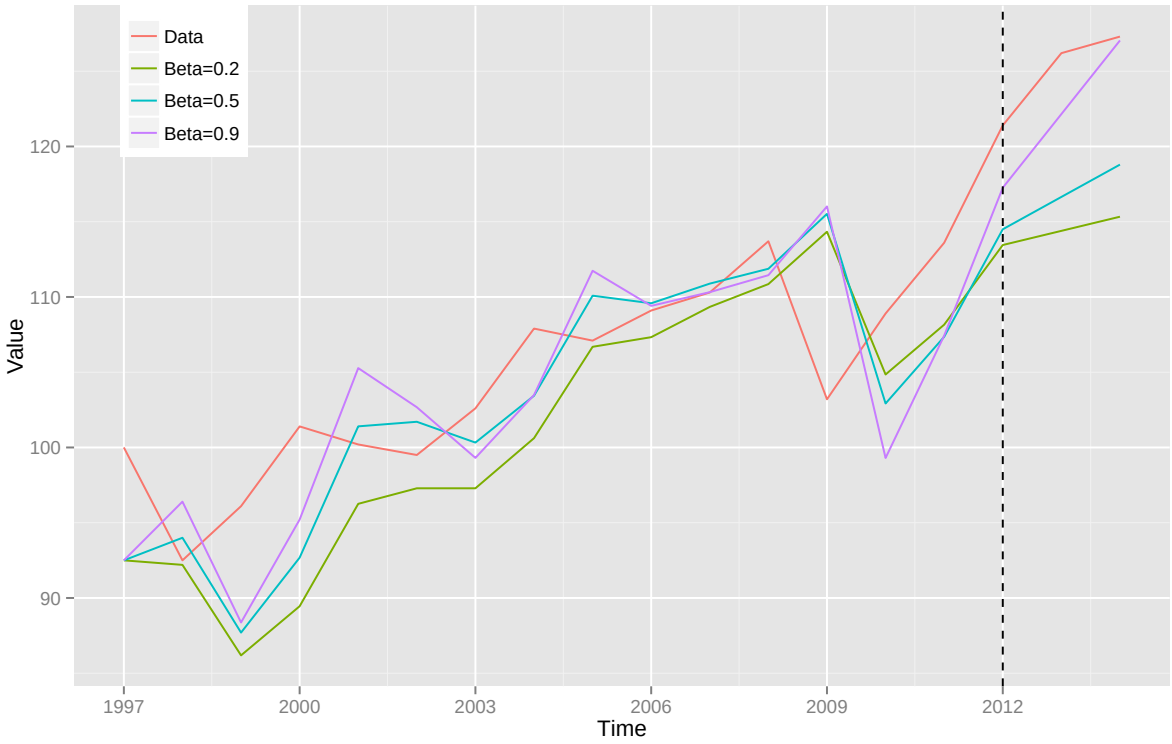
Пример 4.

Для второго примера возьмем годовой показатель индекса пассажирооборота транспорта общественного пользования с 1997 по 2014 год, где значение 1997 года взято за 100 и данные на остальные года пересчитаны относительно этого года. Прогноз будем производить на 3 периода вперед на основе данных до 2011 года включительно. Сглаженные ряды и прогнозы для различных значений параемтра β изобразим на графике:

```
pass <- sophisthse("TRP_Y_PASS_DIRI")

f1 <- holt(ts(pass, start=1997, end=2011),
           alpha=0.8, beta = 0.2, initial="simple", h=3)
f2 <- holt(ts(pass, start=1997, end=2011),
           alpha=0.8, beta = 0.5, initial="simple", h=3)
f3 <- holt(ts(pass, start=1997, end=2011),
           alpha=0.8, beta = 0.9, initial="simple", h=3)

y1 <- c(f1$fitted, f1$mean)
y2 <- c(f2$fitted, f2$mean)
y3 <- c(f3$fitted, f3$mean)

df <- data.frame(x=1997:2014, y=pass$TRP_Y_PASS_DIRI,
                 y1=y1, y2=y2, y3=y3)
df_melted <- melt(df, value.name ="y",
                  measure.vars=c("y","y1","y2","y3") )
```

```
qplot(data=df_melted, x=x, y=y, color=variable, geom="line")+
  labs(list(x = "Time", y = "Value"))+
  ggtitle("Figure 4. Holt's method.")+
  geom_vline(xintercept = 2012, linetype="dashed", color="black")+
  scale_x_continuous(breaks=seq(1997, 2014, 3))+
  scale_colour_discrete(
    name="",
    breaks=c("y", "y1", "y2","y3"),
    labels=c("Data", "Beta=0.2", "Beta=0.5","Beta=0.9"))+
   theme(plot.title = element_text(size = 12, vjust = 2),
         legend.position = c(0.1, 0.9))
```

Figure 4. Holt's method.



Мы можем видеть, что благодаря тому, что данный метод учитывает наличие тренда, он более точно может предсказать поведение ряда. Но прогноз был бы куда менее точным, если бы произошла смена тренда, например, в связи с сезонностью используемых данных.

2.5 Модель Хольта-Уинтерса

Более сложная, но в то же время и более точная в некоторых случаях модель, которая позволяет учитывать сезонность данных. В ней содержится уже 3 сглаживающих уравнения. Помимо уравнений, которые отвечают за уровень и тренд, появляется уравнение сезонности, которое мы будем обозначать s_t.Соответсивенно, появляется третий сглаживающий параметр, который мы обозначим γ, а также n - число сезонов в году. Например, n=12, если данные имеют ежемесячную сезонность.

Сезонность представляет собой влияние внешних факоров на данные с определенной известной периодичностью. Разделяют два типа сезонности - аддитивную и мультипликативную. В связи с этим модель Хольта-Уинтерса имеет две интерпретации.

2.5.1 Модель Хольта-Уинтерса с аддитивной сезонностью.

Аддитивная сезонность имеет место, когда сезонные отклонения данных примерно постоянны относительно изменяющегося среднего значения на протяжении всего наблюдения.В этом случае сезонная компонента выражается в абсолютных значениях и прибавляется к основному уравнению.

$$F_{t+h} = l_t + hb_t + s_{t+h-n} \tag{8}$$

$$l_t = \alpha(Y_t - s_{t-n}) + (1-\alpha)(l_{t-1} + b_{t-1}) \tag{9}$$

$$b_t = \beta(l_t - l_{t-1}) + (1-\beta)b_{t-1} \tag{10}$$

$$s_t = \gamma(Y_t - l_{t-1} - b_{t-1}) + (1-\gamma)s_{t-n} \tag{11}$$

Новое уравнение в модели (11), отвечающее за сезонность, является средневзвешанным между текушим индексом сезонности (в момент времени t) и индексом сезонности того же периода в прошлом году. Отсюда появляется значение t-n.

Уравнение (11) также можно представить в виде:

$$s_t = \gamma(Y_t - l_t) + (1-\gamma)s_{t-n} \tag{12}$$

Чтобы показать, что уравнения (11) и (12) эдентичны, достаточно подставить в уравнение (12)l_t из модели и назвать константой γ^* значение $\gamma(1-\alpha)$.

Пример 5.

Так как данный метод подразумевает наличие сезонности в данных, возьмем тот же индекс, что и в примере 4, но с квартальными данными. Аналогично, прогноз будем строить на 3 года вперед на основе данных до 2011 года включительно.

```
pass <- sophisthse("TRP_Q_PASS_DIRI")

h1 <- HoltWinters(
  ts(pass, start=c(1997,1), end=c(2011,4),frequency = 4),
  alpha=0.8, beta = 0.8, gamma=0.2, seasonal = 'additive')

h2 <- HoltWinters(
  ts(pass, start=c(1997,1), end=c(2011,4),frequency = 4),
  alpha=0.8, beta = 0.8, gamma=0.5, seasonal = 'additive')
```

```
h3 <- HoltWinters(
  ts(pass, start=c(1997,1), end=c(2011,4),frequency = 4),
  alpha=0.8, beta = 0.8, gamma=0.9, seasonal = 'additive')
```

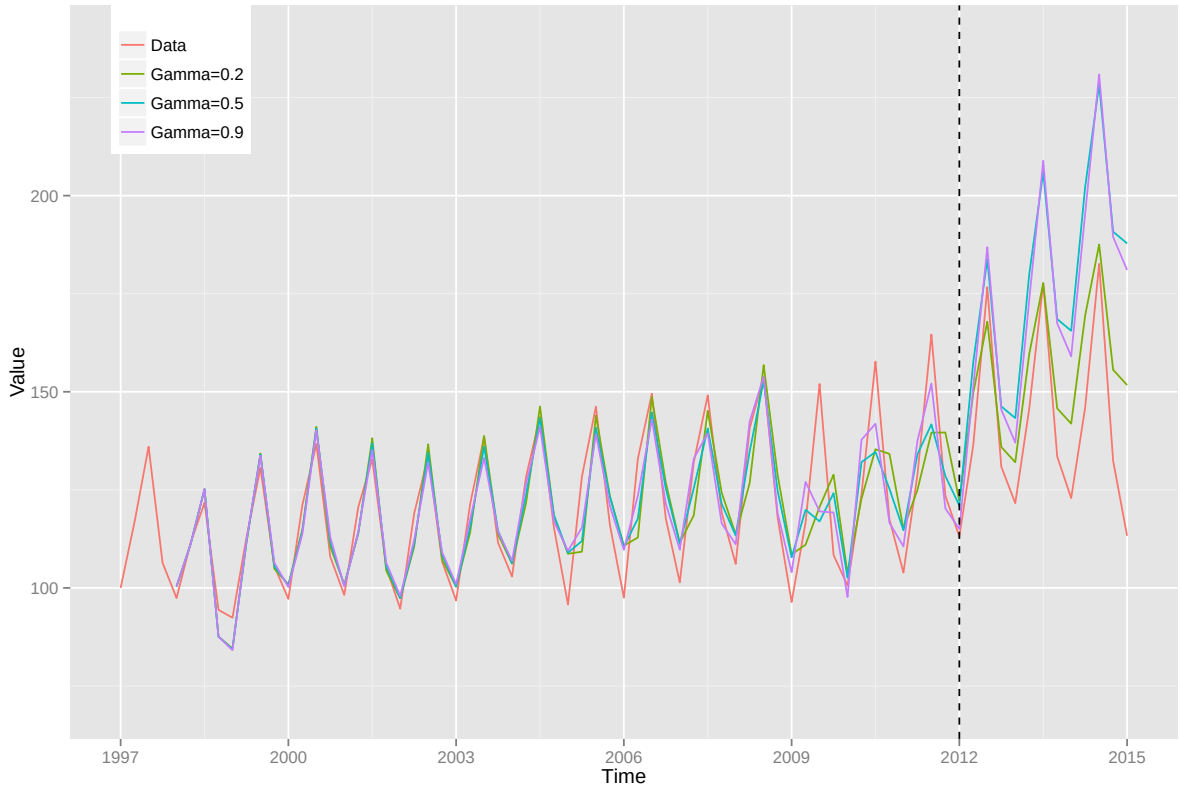
Заметим, с помощью функции HoltWinters() создается только сам сглаженный ряд. То есть для получения прогноза необходимо применить к сглаженному ряду функцию predict().

```
f1<- predict(h1, n.ahead = 13)
f2<- predict(h2, n.ahead = 13)
f3<- predict(h3, n.ahead = 13)

y1<-c(rep(0,4), h1$fitted[,1],f1 )
y2<-c(rep(0,4), h2$fitted[,1],f2 )
y3<-c(rep(0,4), h3$fitted[,1],f3)

df <- data.frame(x=1:73, y=pass$TRP_Q_PASS_DIRI,
                 y1=y1, y2=y2, y3=y3 )
df_melted <- melt(df, value.name ="y",
                  measure.vars=c("y","y1","y2","y3") )
```

```
qplot(data=df_melted, x=x, y=y, color=variable, geom="line")+
  labs(list(title = "Figure 5.Holt-Winters additive method.",
            x = "Time", y = "Value"))+
  geom_vline(xintercept=61, linetype="dashed", color="black")+
  ylim(70,240)+
  scale_x_continuous(breaks=seq(1, 73, 12),
                     labels=seq(1997, 2015, 3))+
  scale_colour_discrete(
    name="",
    breaks=c("y", "y1", "y2","y3"),
    labels=c("Data", "Gamma=0.2", "Gamma=0.5","Gamma=0.9"))+
   theme(plot.title = element_text(size = 12, vjust = 2),
```

```
          legend.position = c(0.1, 0.9))
```

Figure 5.Holt−Winters additive method.



2.5.2 Модель Хольта-Уинтерса с мультипликативной сезонностью.

Мультипликативная сезонность означает, что сезонное отклонение данных изменяется пропорцианально изменению самих данных. И в этом случае сезонная компонента выражается в процентах (или долях) и прогноз получается при помощи умножения основного уравнения на нее.

$$F_{t+h} = (l_t + hb_t)s_{t+h-n} \tag{13}$$

$$l_t = \alpha\frac{Y_t}{s_{t-n}} + (1-\alpha)(l_{t-1} + b_{t-1}) \tag{14}$$

$$b_t = \beta(l_t - l_{t-1}) + (1-\beta)b_{t-1} \tag{15}$$

$$s_t = \gamma\frac{Y_t}{l_{t-1} + b_{t-1}} + (1-\gamma)(s_{t-n}) \tag{16}$$

Также, как и в случае с аддитивной моделью, уравнение сезонности (16) можно записать в другом, более коротком, виде:

$$s_t = \gamma\frac{Y_t}{l_t} + (1-\gamma)(s_{t-n}) \tag{17}$$

Пример 6.

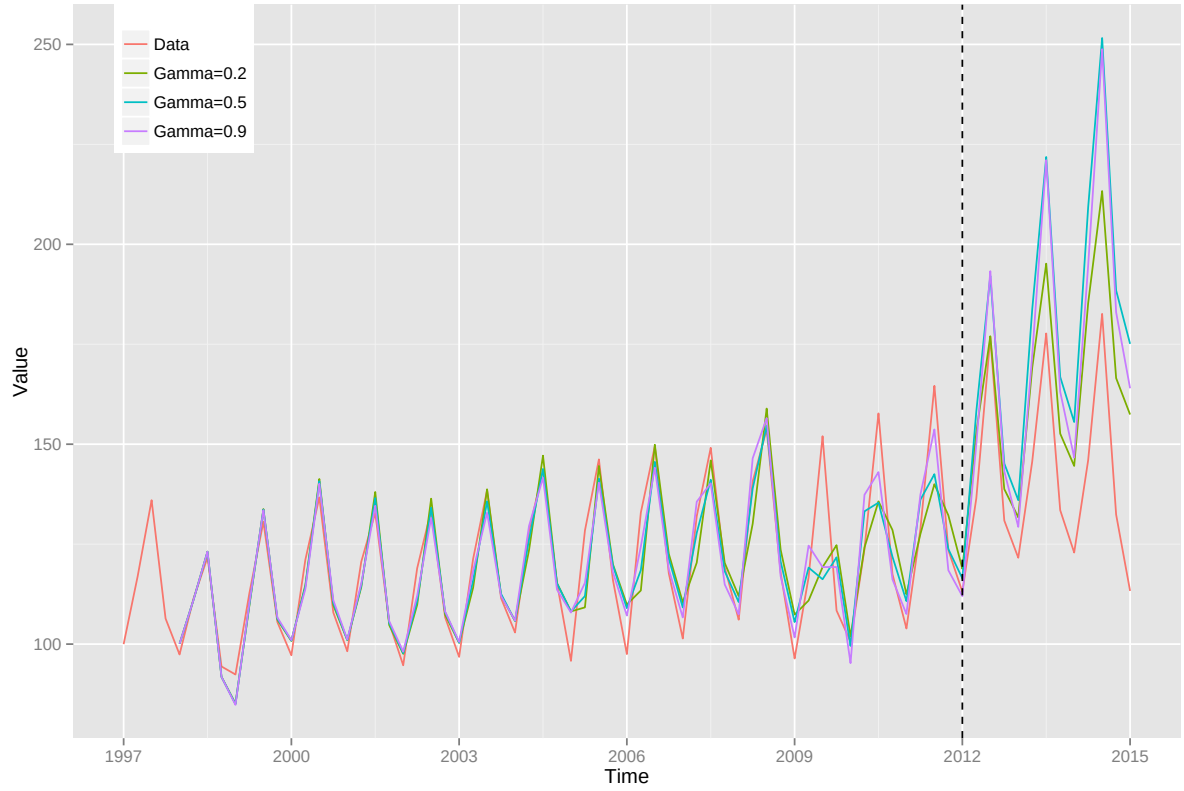
```
pass <- sophisthse("TRP_Q_PASS_DIRI")

h1 <- HoltWinters(
  ts(pass, start=c(1997,1), end=c(2011,4),frequency = 4),
  alpha=0.8, beta = 0.8, gamma=0.2, seasonal = 'mult')

h2 <- HoltWinters(
  ts(pass, start=c(1997,1), end=c(2011,4),frequency = 4),
  alpha=0.8, beta = 0.8, gamma=0.5, seasonal = 'mult')

h3 <- HoltWinters(
  ts(pass, start=c(1997,1), end=c(2011,4),frequency = 4),
  alpha=0.8, beta = 0.8, gamma=0.9, seasonal = 'mult')
```

```
f1<- predict(h1, n.ahead = 13)
f2<- predict(h2, n.ahead = 13)
f3<- predict(h3, n.ahead = 13)

y1<-c(rep(NA,4), h1$fitted[,1],f1 )
y2<-c(rep(NA,4), h2$fitted[,1],f2 )
y3<-c(rep(NA,4), h3$fitted[,1],f3)

df <- data.frame(x=1:73, y=pass$TRP_Q_PASS_DIRI,
                 y1=y1, y2=y2, y3=y3 )
df_melted <- melt(df, value.name ="y",
                  measure.vars=c("y","y1","y2","y3") )
```

```
qplot(data=df_melted, x=x, y=y, color=variable, geom="line")+
  labs(list(title = "Рисунок 6. Holt-Winters multiplicative method",
```

```
          x = "Time", y = "Value"))+
 geom_vline(xintercept = 61, linetype="dashed", color="black")+
 scale_x_continuous(breaks=seq(1, 73, 12),
                    labels=seq(1997, 2015, 3))+
 scale_colour_discrete(
   name="",
   breaks=c("y", "y1", "y2","y3"),
   labels=c("Data", "Gamma=0.2", "Gamma=0.5","Gamma=0.9"))+
  theme(plot.title = element_text(size = 12, vjust = 2),
        legend.position = c(0.1, 0.9))
```

Рисунок 6. Holt−Winters multiplicative method



Оба эти метода не имеют большого колличества различий. По графикам видно, что прогнозы в обоих случаях достаточно точны и хорошо предугадывают сезонные изменения тренда. Но разница между данными все же имеется. Так, можно видеть, что прогноз при $\gamma = 0.2$ в модели с аддитивной сезонностью гораздо точнее аналогичного во второй модели.

2.6 Тета-метод

Этот метод был представлен в 2000 году('The theta model: a decomposition approach to forecasting', Assimakopoulos, Nikolopoulos, 2000).

Метод заключается в том, что для прогнозирования исходного ряда Y_t рассчитывается такой ряд $S_{t,\theta}$, что для него выполняется следующее равенство:

$$\Delta^2 S_{t,\theta} = \theta \cdot \Delta^2 Y_t \tag{18}$$

$\Delta^2 Y_t$ - вторая разность или ускорение ряда, которая вычисляется следующим образом:

$$\Delta^2 Y_t = \Delta Y_{t+1} - \Delta Y_t = Y_{t+2} - 2Y_{t+1} + Y_t \tag{19}$$

Полученный из уравнения (18) ряд $S_{t,\theta}$ называют тета-линией:

$$S_{t,\theta} = a_\theta + b_\theta(t-1) + \theta Y_t \tag{20}$$

где a_θ и b_θ - константы.

Заметим, что для первых двух элементов нового ряда нельзя определить вторую разность, поэтому они находятся путем минимизации суммы квадратов расстояний до исходных данных:

$$\sum_{i=1}^{t}(Y_t - S_{t,\theta})^2 = \sum_{i=1}^{t}[(1-\theta)Y_t - a_\theta - b_\theta(t-1)]^2 \to \min_{a_\theta, b_\theta} \tag{21}$$

Непосредственно для прогнозирования используют средневзвешанный прогноз для $S_{t,\theta}$ при различных значениях θ. Авторы данного метода используют две тета-линии (при $\theta = 0$ и $\theta = 2$), и тогда прогноз на основании n наблюдений на h периодов вперед выглядит следующим образом:

$$F_n(h) = \frac{1}{2}[\hat{S}_{n,0}(h) + \hat{S}_{n,2}(h)] \tag{22}$$

$\hat{S}_{n,0}(h)$ представляет собой линейную часть уравнения (20) и является линейным трендом, расчитанным на основе метода наименьших квадратов (см. (21)), а $\hat{S}_{n,2}(h)$ получается при экспоненциальном сглаживании ряда $S_{t,2}$

Пример 7.

Для тех же данных по годовому показателю индекса пассажирооборота транспорта общественного пользования рассчитаем прогноз на три периода вперед тета-методом. Для этого воспользуемся функцией thetaf(). Заметим, что параметр α для экспоненциального сглаживания тета-линии $S_{t,2}$ подбирается автоматически методом наименьших квадратов.

```
pass <- sophisthse("TRP_y_PASS_DIRI")


y1 <- thetaf(ts(pass, start=1997, end=2011), h=3)
df_melted <- melt(
  data.frame(x=1997:2014, y=pass$TRP_Y_PASS_DIRI,
             y1=c(y1$fitted,y1$mean)),
  value.name ="y", measure.vars=c("y","y1"))
```

```
qplot(data=df_melted, x=x, y=y, color=variable, geom="line")+
  labs(list(title = "Figure 7.Theta method.", x = "Time", y = "Value"))+
  scale_x_continuous(breaks=seq(1997,2014, 3)) +
  geom_vline(xintercept = 2012, linetype="dashed", color="black")+
  scale_colour_discrete(name="",
                        breaks=c("y", "y1"),
                        labels=c("Data","Smoothed/Forecast"))+
   theme(plot.title = element_text(size = 12, vjust = 2),
         legend.position = c(0.1, 0.9))
```

Figure 7.Theta method.

3 Методы оценки точности прогноза

Для того, чтобы сравнить различные методы прогнозирования между собой, необходимо выбрать критерий сравнения. Таких критериев существует достаточно много, но в данной работе будет рассмотрено три наиболее часто используемые.

RMSE

RMSE (Root Mean Squared Error) или корень квадратный из средней квадратичной ошибки можно найти по формуле:

$$RMSE = \sqrt{\frac{\sum(F_t - Y_t)^2}{n}} \tag{23}$$

В связи с тем, что все ошибки сначала возводятся в квадрат, а потом усредняются, получается, что чем больше ошибка, тем больший вес она имеет (например, одна ошибка в 10 единиц весит больше, чем две по 5). Это нельзя назвать недостатком данного метода, так как это может быть полезным, если одна крупная ошибка в конктретном случае гораздо хуже нескольких более мелких.

MAE

MAE (mean absolut error) или средняя абсолютная ошибка определяется следующим образом:

$$MAE = \frac{\sum|F_t - Y_t|}{n} \tag{24}$$

Благодаря модулю все ошибки в данном случае имеют одинаковый вес.

MAPE

MAPE (Mean Absolute Percent Error) или средняя абсолютная процентная ошибка находится по формуле:

$$MAPE = \frac{\sum|\frac{Y_t - F_t}{Y_t}|}{n} \cdot 100 \tag{25}$$

Главное достоинство этого метода заключается в том, что он измеряет ошибку в процентах от исходных данных, что, во-первых, является более наглядным показателем и, во-вторых, позволяет без проблем сравнивать между собой ошибки для различных временных рядов.

4 Данные

Для сравнения представленных методов по их прогнозной силе использовалось 16 временных рядов, которые можно условно разделить на четыре группы.

Данные	Время между наблюдениями
1.Производство товаров и услуг (Industry)	
Индекс обрабатывающего произвводства	Квартал
Индекс производства Добыча сырой нефти	Месяц
Индекс реального объема сельскохозяйственного производства	Год
Ввод в действие жилых домов	Квартал
2.Макроэкономические данные (Macro)	
Расход ВВП на конечное потребление	Год
Чистый экспорт	Год
Индекс реального ВВП	Квартал
Индекс потребительских цен	Квартал
3.Финансовые данные (FInance)	
Официвльный курс доллара	Месяц
Акции аэрофлота	Месяц
Индекс ММВБ	Месяц
Сбербанк	Месяц
4.Население и трудовые ресурсы (Demographic)	
Численность населения	Год
Коэффициент рождаемости	Год
Коэффициент смертности	Год
Уровень безработицы	Квартал

Таблица 1: Ряды, используемые в исследовании

При прогнозировании параметры (там где необходимо) вычисляются при помощи минимизации суммы квадратов отклонений (SSE) сглаженного ряда от исходного. Горизонт прогноза зависит от частоты имеющихся наблюдений:

Время между наблюдениями	Горизонт прогноза
Год	3
Квартал	5
Месяц	17

Таблица 2: Используемый горизонт прогноза

5 Расчеты

Приступим непосредственно к сравнению нестатисчическх методов. В первую очередь нам необходимо скачать выбранные временные ряды. Для этого создадим таблицу, в которой будет находится вся необходимая информация о них.

```
names <- c("IP_DEA_Q","ECOG2","AGR_Y_DIRI", "CONSTR_Q_NAT", "GDPS_Y",
           "GDPS_Y", "GDPEA_Q", "CPI_Q_CHI", "USDCB", "AFLT", "MICEX",
           "MTSS", "POPNUM_Y", "POPFER_Y", "POPMOR_Y", "UNEMPL_Q_SH")

series_info <- data_frame(names=names, source="sophist",
                          freq=12, col_number=1)
series_info$source[9:12] <- "finam"
series_info$freq[c(3,5,6,13,14,15)]<- 1
series_info$freq[c(1,4,7,8,16)]<- 4
series_info$col_number[2]<-2
series_info$col_number[6]<-8
```

Все данные будем скачивать в объект data, который имеет формат list.

```
data <-list()
for (i in 1:16){
 if (series_info$source[i]=="sophist") {
   data[[i]] <- sophisthse(series_info$names[i])
 }
 if (series_info$source[i]=="finam") {
   data[[i]]<-getSymbols(Symbol=series_info$names[i],
                         from="2000-01-01", src="Finam", period="month")
 }}
```

```
finam <- list(USDCB, AFLT, MICEX, MTSS)
for (i in 1:4) {
  data[[i+8]] <- finam[[i]]
}
```

Теперь можно приступить к прогнозированию рядов и подсчету ошибок прогноза. Для удобства разделим ошибки в зависимости от метода их подсчета и подготовим три таблицы данных, в которые они впоследствии будут записываться.

```
error_RMSE <- data_frame(names=names, naive = 0, average = 0,
                         ses = 0, holt = 0, hw_mult =0,hw_add=0, theta=0 )
error_MAE <- data_frame(names=names, naive = 0, average = 0,
                         ses = 0, holt = 0, hw_mult =0,hw_add=0, theta=0 )
error_MAPE <- data_frame(names=names, naive = 0, average = 0,
                         ses = 0, holt = 0, hw_mult =0,hw_add=0, theta=0 )
```

Так как в большинстве случаев скачиваются не отдельные временные ряды, а целые таблицы, нам понадобится вектор, содержащий названия именно тех рядов, которые были выбраны для анализа.

```
col_names <- c("IP_DEA_Q ", "ECOG2", "AGR_Y_DIRI", "CONSTR_Q_NAT",
               "GDPS_FC_Y", "GDPS_NEXP_Y", "GDPEA_Q_DIRI", "CPI_Q_CHI",
               "USDCB.Open", "AFLT.Open", "MICEX.Open", "MTSS.Open",
               "POPNUM_Y", "POPFER_Y", "POPMOR_Y", "UNEMPL_Q_SH")
```

При прогнозмровании все 16 рядов, в первую очередь, необходмо разделить на группы, в зависимости от частоты наблюдений. От этого зависит горизон прогноза и, соответственно алгорит расчета. Для каждого ряда расчитывается прогноз семью выбранными методами и считается три типа ошибки.

Заметим, что метод Хольта-Уинтерса подразумевает наличие сезонности в данных, а значит, нескольких наблюдений в течении года. Поэтому для этих методов не расчитывался прогноз для рядов с частотой в 1 год.

```
for (i in 1:16){
  if(series_info$freq[i]=="1"){
    table <- data[[i]]
```

```
  data_ts <- ts(table[, series_info$col_number[i]])
  length <- length(table[, series_info$col_number[i]])-3

  for (j in 1:7){
    if(j=="1"){
      forecast <- rep(data_ts[length], 3)}
    if(j=="2"){
      forecast <- rep(mean(data_ts[1:length]), 3)}
    if(j=="3"){
      fitted <- ses(data_ts[1: length], initial="optimal", h = 3)
      forecast <- fitted$mean }
    if(j=="4"){
      fitted <- holt(data_ts[1: length], initial="optimal", h = 3)
      forecast <- fitted$mean}
    if(j=="5"){
      forecast<-rep(NA,3)}
    if(j=="6"){
      forecast<-rep(NA,3) }
    if(j=="7"){
      fitted <- thetaf(data_ts[1: length],  h = 3)
      forecast <- fitted$mean }

  a<-length+1
  b<-length+3
  error_RMSE[i,j+1] <- sqrt(mean (data_ts[a:b]- forecast)^2)
  error_MAE[i,j+1] <- mean(abs(data_ts[a:b] - forecast))
  error_MAPE[i,j+1] <-
    sum(abs((data_ts[a:b]-forecast)/data_ts[a:b]))/length(data_ts[a:b])
}}
if(series_info$freq[i]=="4"){
  table <- data[[i]]
  data_ts <- ts(table[, series_info$col_number[i]], frequency=4)
  length <- length(table[, series_info$col_number[i]])-5
```

```
  for (j in 1:7){
    if(j=="1"){
      forecast <- rep(data_ts[length], 5)}
    if(j=="2"){
      forecast <- rep(mean(data_ts[1:length]), 5) }
    if(j=="3"){
      fitted <- ses(data_ts[1: length], initial="optimal", h = 5)
      forecast <- fitted$mean }
    if(j=="4"){
      fitted <- holt(data_ts[1: length], initial="optimal", h = 5)
      forecast <- fitted$mean}
    if(j=="5"){
      fitted <- HoltWinters(data_ts, seasonal = "mult")
      forecast<- predict(fitted, n.ahead = 5) }
    if(j=="6"){
      fitted <- HoltWinters(data_ts, seasonal = "additive")
      forecast<- predict(fitted, n.ahead = 5) }
    if(j=="7"){
      fitted <- thetaf(data_ts[1: length],  h = 5)
      forecast <- fitted$mean}

  a<-length+1
  b<-length+5
  error_RMSE[i,j+1] <- sqrt(mean (data_ts[a:b]- forecast)^2)
  error_MAE[i,j+1] <- mean(abs(data_ts[a:b] - forecast))
  error_MAPE[i,j+1] <-
    sum(abs((data_ts[a:b]-forecast)/data_ts[a:b]))/length(data_ts[a:b])
}}
if(series_info$freq[i]=="12"){
  table <- data[[i]]
  data_ts <- ts(table[, series_info$col_number[i]], frequency=12)
  length <- length(table[, series_info$col_number[i]])-17
```

```
    for (j in 1:7){
      if(j=="1"){
        forecast <- rep(data_ts[length], 17) }
      if(j=="2"){
        forecast <- rep(mean(data_ts[1:length]), 17)}
      if(j=="3"){
        fitted <- ses(data_ts[1: length], initial="optimal", h = 17)
        forecast <- fitted$mean }
      if(j=="4"){
        fitted <- holt(data_ts[1: length], initial="optimal", h = 17)
        forecast <- fitted$mean}
      if(j=="5"){
        fitted <- HoltWinters(data_ts, seasonal = "mult")
        forecast<- predict(fitted, n.ahead = 17)}
      if(j=="6"){
        fitted <- HoltWinters(data_ts, seasonal = "additive")
        forecast<- predict(fitted, n.ahead = 17)}
      if(j=="7"){
        fitted <- thetaf(data_ts[1: length],  h = 17)
        forecast <- fitted$mean }

    a<-length+1
    b<-length+17
    error_RMSE[i,j+1] <- sqrt(mean (data_ts[a:b]- forecast)^2)
    error_MAE[i,j+1] <- mean(abs(data_ts[a:b] - forecast))
    error_MAPE[i,j+1] <-
      sum(abs((data_ts[a:b]-forecast)/data_ts[a:b]))/length(data_ts[a:b])
  }}
}
```

Наконец, необходимо сравнить ошибки между собой. Будем сранивать ошибку каждого из семи методов по всем рядам и выбирать наименьшую.Результаты сравнения занесем в таблицу results.

```
series_names<- c("Обрабатывающее пр-во", "Добыча нефти","С/х пр-во",
"Ввод жилых домов","Потребление","Чистый экспорт", "Реальный ВВП",
"Потребительские цены", "Курс доллара", "Акции аэрофлота","ММВБ" ,
"Сбербанк" ,"Численность населения", "Рождаем
results <- data.frame(Series=series_names, Frequency=12,RMSE=rep(0,16),
                      MAE=rep(0,16), MAPE=rep(0,16))
results$Frequency[c(1,4,7,8,16)]<- 4
results$Frequency[c(3,5,6,13,14,15)] <- 1
```

```
for (i in 1:16){
  a <- which.min(error_RMSE[i,] )
  b <- which.min(error_MAE[i,] )
  c <- which.min(error_MAPE[i,] )
  results$RMSE[i]<- names(a)
  results$MAE[i]<- names(b)
  results$MAPE[i]<- names(c)
}
```

```
xtable(results, caption="Резуль
```

	Series	Frequency	RMSE	MAE	MAPE
1	Обрабатывающее пр-во	4.00	hw_add	ses	ses
2	Добыча нефти	12.00	ses	theta	theta
3	С/х пр-во	1.00	naive	holt	holt
4	Ввод жилых домов	4.00	hw_mult	theta	holt
5	Потребление	1.00	holt	holt	holt
6	Чистый экспорт	1.00	ses	holt	holt
7	Реальный ВВП	4.00	hw_mult	ses	ses
8	Потребительские цены	4.00	naive	naive	naive
9	Курс доллара	12.00	hw_mult	hw_mult	holt
10	Акции аэрофлота	12.00	hw_add	average	average
11	ММВБ	12.00	ses	theta	ses
12	Сбербанк	12.00	hw_add	hw_add	hw_add
13	Численность населения	1.00	holt	holt	holt
14	Рождаемость	1.00	holt	holt	holt
15	Смертность	1.00	naive	naive	naive
16	Безработица	4.00	hw_add	hw_add	hw_add

Таблица 3: Результаты

Исходя из данной таблицы можно сделать несколько выводов.

Во-первых, видно, что более сложные методы не всегда дают более точный прогноз. Например, для данных по уровню сметрности (ряд 15) лучший прогноз показал самый простой из выбранныхх методов. Аналогичная ситуация с показателем потребительских цен (ряд 8).

Во-вторых, точность прогноза в некоторой степени зависит от метода, с помощью которого она оценивается. Так, в случае с показателем количества введенных жилых домов (ряд 4), для всех трех методов оценки разные методы показали меньшую ошибку.

6 Заключение

Целью данной работы было описание и сравнение по прогнозной силе нестатистических методов прогнозирования временных рядов. Для анализа было выбрано семь методов различного уровня сложности и описан механизм, с помощью которого производится расчет будущих показателей в каждом из них.

Исследовательская часть работы заключалась в применении выбраных нестатистических методов к реальным данным и сравнении точности полученных прогнозов в зависимости от метода и типа данных. С этой целью было выбрано три метода оценки точности прогноза, описан способ их подсчета и основные особенности.

На основе полученных результатов удалось сделать вывод о том, что самые простые методы прогнозирования могут давать более точные результаты, чем более сложные для некоторых видов данных. Однако оценка точности может значительно зависить от метода, при помощи которого она проводится. Конечно, стоит отметить, что для выявления закономерностей и определения наиболее точных методов необходимо использовать куда большее количество данных, как, например в M и M3 соревнованиях, где анализ проводился на основе 1001 и 3003 рядов соответственно.

7 Список литературы

1. Rob J Hyndman, George Athanasopoulos. Forecasting: principles and practice, 2012.

2. V. Assimakopoulos, K. Nikolopoulos. The theta model: a decomposition approach to forecasting. International Journal of Forecasting, 16, 521-530.

3. Rob J Hyndman, Anne B. Koehler. Another look at measures of forecast accuracy. International Journal of Forecasting, 22, 679-688.

4. Michele Hibon, Spyros Makridakis. The M3-Competition: results, conclusions and implications. International Journal of Forecasting, 16, 451-476.

5. Rob J Hyndman, Baki Billah. Unmasking the Theta method. International Journal of Forecasting, 19, 287-290.

6. Chris Chatfield, Mohammad Yar. Holt-Winters Forecasting: Some Practical Issues.Journal of the Royal Statistical Society. Series D (The Statistician), Vol. 37, No. 2, 129-140.

Источники данных (электронные ресурсы)

1. http://sophist.hse.ru/

2. http://www.finam.ru/