INDR 450/550

Spring 2022

Lecture 10: Regression for Time Series

March 16, 2022

Fikri Karaesmen

Announcements

- Class Exercise at the end of lecture today. If you are participating online, please upload your document under Course Contents/Class Exercises
- HW 1 due date approaching.
- HW 2 available soon.

- The first three labs were uploaded. Please follow them.
  - Next HW based on lab2

A few notes and corrections

- Homework: the data is not very pretty. Simpler models may perform better.
  - Please try to understand why.

Summary of ACF and PACF patterns for simple AR and MA models

Process	ACF	PACF
AR(1)	Exponential decay: on positive side if $\phi_1 > 0$ and alternating in sign starting on negative side if $\phi_1 < 0$.	Spike at lag 1, then cuts off to zero: spike positive if $\phi_1 > 0$, negative if $\phi_1 < 0$.
AR(p)	Exponential decay or damped sine-wave. The exact pattern depends on the signs and sizes of $\phi_1, \dots, \phi_p$.	Spikes at lags 1 to p, then cuts off to zero.
MA(1)	Spike at lag 1 then cuts off to zero: spike positive if $\theta_1 < 0$, negative if $\theta_1 > 0$.	Exponential decay: on negative side if $\theta_1 > 0$ and alternating in sign starting on positive side if $\theta_1 < 0$.
MA(q)	Spikes at lags 1 to q, then cuts off to zero.	Exponential decay or damped sine-wave. The exact pattern depends on the signs and sizes of $\theta_1, \dots, \theta_q$.

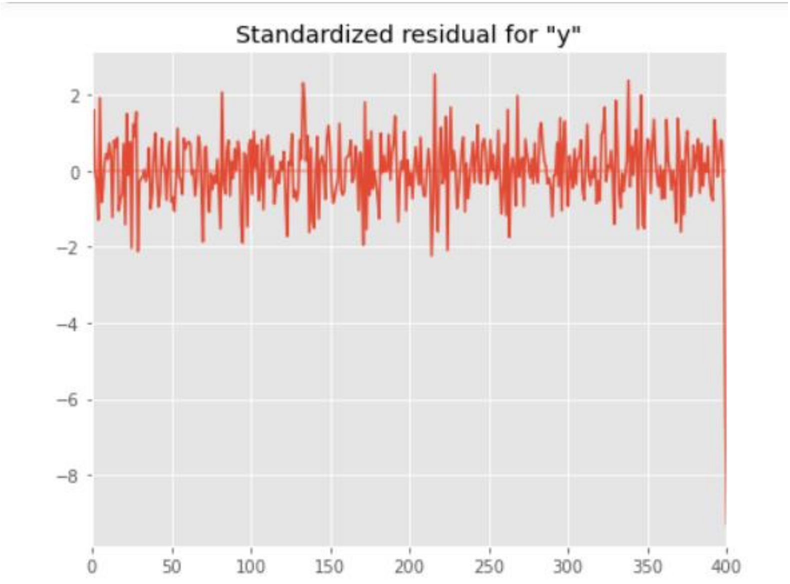
Table 7-2: *Expected patterns in the ACF and PACF for simple AR and MA models.*

Please note that the MA-terms are defined with a negative sign in this Reference. This is why the signs are reversed in the examples in Lab 3.

$$Y_t = c - \theta_1 \varepsilon_{t-1} - \theta_2\, \varepsilon_{t-2} + \varepsilon_t$$

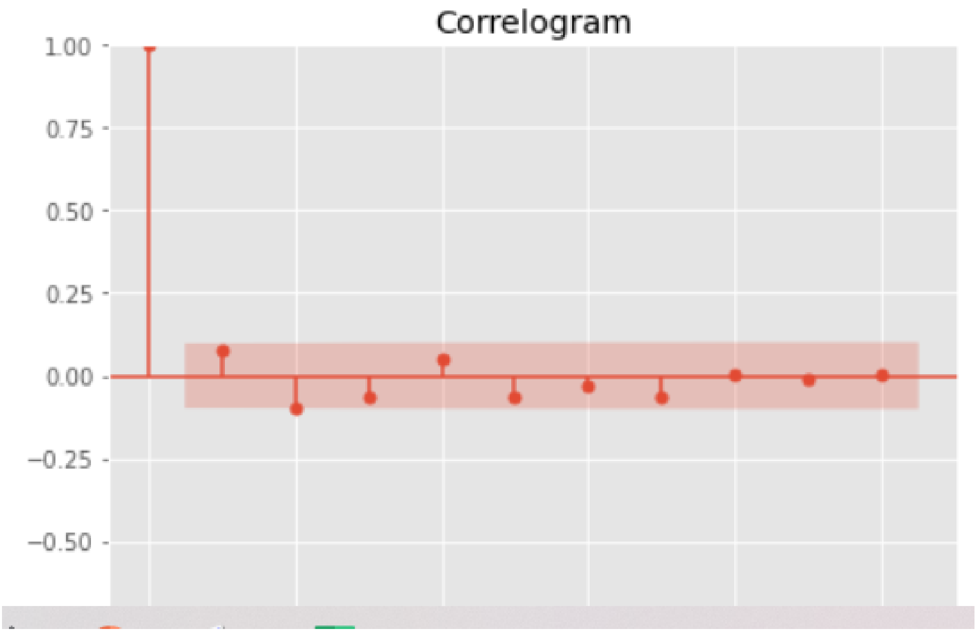
Makridakis, Wheelwright and Hyndman (1997)

ARIMA Framework: Residuals



- Recall that we want our forecasts to be unbiased: $E[\hat{Y}_t] = E[Y_t]$. But this can only be checked if we know the theoretical model.
- But we can check unbiasedness from the residuals of the fitted model $e_t = y_t - \hat{y}_t$.
- The average of the residuals must be zero (i.e. Statistically indistinguishable from zero).
- In this example, we have an unbiased forecast.

ARIMA Framework: Residuals



- We also want that the residuals do not have any auto-correlation remaining. Otherwise, the ARIMA model would be insufficient to explain the auto-correlation structure.
- In this example, the residuals do not have any auto-correlation. The fitted model is sufficient to explain the auto-correlation structure.

Regression for Time Series

- Consider the following linear model:

$$y_t = \beta_0 + \beta_1 x_{1t} + \beta_2 x_{2t} + \ldots + \beta_n x_{nt} + \epsilon_t$$

- y_t is the forecast and x_{kt} are the predictors.

Note that we can use other time series on the right hand side of a regression. For instance, to predict the sales of dishwashers, we can take as a predictor the sales of refrigerators from one month Ago.

Regression for Time Series

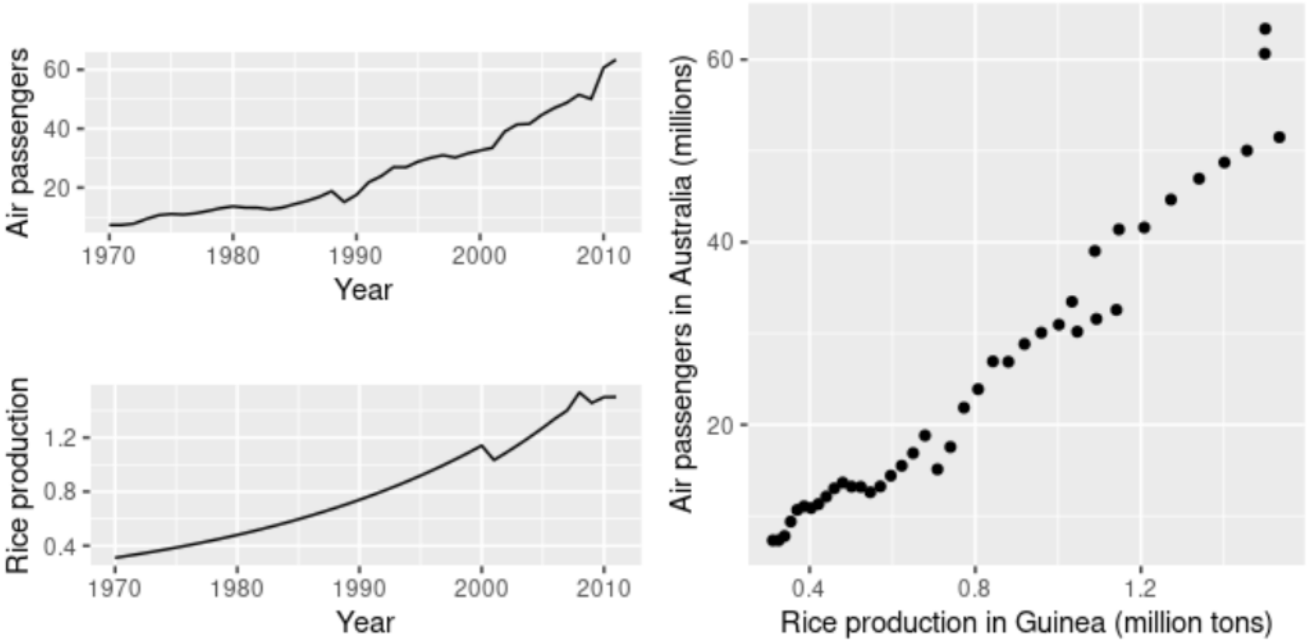


Figure 5.12: Trending time series data can appear to be related, as shown in this example where air passengers in Australia are regressed against rice production in Guinea.

But it's not a good idea to predict the number of air passengers by using as a predictor rice production in Guinea! They are correlated but unrelated, the prediction will be completely off if Guinea allocates additional land to rice farmers or if there's a drought.

Regression for Time Series

- Linear regression is a general tool that looks for a linear relationship between a response and predictors.
- We have observations at different levels of the predictors and the corresponding response.
- The goal is to have predictions for the response that will be generated by so far unobserved levels of the predictors.

- We'll look into time series data where the data is the time series itself. The prediction is then typically a forecast for future demand, prices etc.

Regression for Time Series

- Consider the following linear model:

$$y_t = \beta_0 + \beta_1 x_{1t} + \beta_2 x_{2t} + \ldots + \beta_n x_{nt} + \epsilon_t$$

- y_t is the forecast and x_{kt} are the predictors.
- We are therefore looking for a linear relationship between the predictors and the response (the forecast).
- Note that in the setting of forecasting, this is somewhat different than designing a controlled experiment where we can control the levels of the predictors. The predictors that are available to us cannot be controlled in general.

Regression for Time Series: Basic Predictors: Google Share Price

- The Google Share Price Data has a strong trend.
- Let's try a simple trend based regression. Call this Model 1.

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

	Price	Day
1	2064.879883	1
2	2070.860107	2
3	2095.169922	3
4	2031.359985	4
5	2036.859985	5

Regression for Time Series: Basic Predictors: Google Share Price

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

In [3]:
```
lm = sm.OLS.from_formula('Price ~ Day', df)
result = lm.fit()
result.summary()
```

Out[3]:

OLS Regression Results

Dep. Variable:	Price	**R-squared:**	0.712
Model:	OLS	**Adj. R-squared:**	0.711
Method:	Least Squares	**F-statistic:**	619.5
Date:	Tue, 08 Mar 2022	**Prob (F-statistic):**	9.84e-70
Time:	17:39:05	**Log-Likelihood:**	-1633.4
No. Observations:	253	**AIC:**	3271.
Df Residuals:	251	**BIC:**	3278.
Df Model:	1		
Covariance Type:	nonrobust		

	coef	**std err**	**t**	**P>|t|**	**[0.025**	**0.975]**
Intercept	2197.6246	19.501	112.694	0.000	2159.218	2236.031
Day	3.3131	0.133	24.890	0.000	3.051	3.575

Omnibus:	13.705	**Durbin-Watson:**	0.068
Prob(Omnibus):	0.001	**Jarque-Bera (JB):**	15.010
Skew:	-0.590	**Prob(JB):**	0.000550
Kurtosis:	2.818	**Cond. No.**	294.

$$\hat{\beta}_0 = 2197.62, \hat{\beta}_0 = 3.31$$

Regression for Time Series: Basic Predictors: Google Share Price

- Since the execution is very easy, we are tempted to try other predictors, let us try:

$$y_t = \beta_0 + \beta_1 t + \beta_2 \sqrt{t} + \beta_3 t^2 + \epsilon_t$$

Out[2]:

	Price	Day	Sqrtd	Sqrd
1	2064.879883	1	1.000000	1
2	2070.860107	2	1.414214	4
3	2095.169922	3	1.732051	9
4	2031.359985	4	2.000000	16
5	2036.859985	5	2.236068	25

In [12]:
```
lm2 = sm.OLS.from_formula('Price ~ Day + Sqrtd+ Sqrd', df)
result2 = lm2.fit()
```

Let us call this model Model 2.

Regression for Time Series: Basic Predictors: Google Share Price

$$y_t = \beta_0 + \beta_1 t + \beta_2 \sqrt{t} + \beta_3 t^2 + \epsilon_t$$

13]:

OLS Regression Results

Dep. Variable:	Price	**R-squared:**	0.937
Model:	OLS	**Adj. R-squared:**	0.936
Method:	Least Squares	**F-statistic:**	1236.
Date:	Tue, 08 Mar 2022	**Prob (F-statistic):**	3.40e-149
Time:	12:17:45	**Log-Likelihood:**	-1440.8
No. Observations:	253	**AIC:**	2890.
Df Residuals:	249	**BIC:**	2904.
Df Model:	3		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
Intercept	2207.3790	43.714	50.496	0.000	2121.282	2293.476
Day	19.8417	1.304	15.219	0.000	17.274	22.409
Sqrtd	-111.2141	14.968	-7.430	0.000	-140.695	-81.733
Sqrd	-0.0432	0.002	-18.802	0.000	-0.048	-0.039

Omnibus:	9.814	**Durbin-Watson:**	0.308
Prob(Omnibus):	0.007	**Jarque-Bera (JB):**	9.833
Skew:	-0.450	**Prob(JB):**	0.00733
Kurtosis:	3.352	**Cond. No.**	2.90e+05

Regression for Time Series: Basic Predictors: Google Share Price

$$y_t = \beta_0 + \beta_1 t + \beta_2 \sqrt{t} + \beta_3 t^2 + \epsilon_t$$

	coef	std err	t	P>|t|	[0.025	0.975]
Intercept	2207.3790	43.714	50.496	0.000	2121.282	2293.476
Day	19.8417	1.304	15.219	0.000	17.274	22.409
Sqrtd	-111.2141	14.968	-7.430	0.000	-140.695	-81.733
Sqrd	-0.0432	0.002	-18.802	0.000	-0.048	-0.039

- Note that $\hat{\beta}_2$ and $\hat{\beta}_3$ are also statistically significant.

Regression for Time Series: Basic Predictors: Google Share Price

- We can become even more aggressive and try adding a log term to the model

$$y_t = \beta_0 + \beta_1 t + \beta_2 \sqrt{t} + \beta_3 t^2 + \beta_4 \log(t) + \epsilon_t$$

Out[19]:

OLS Regression Results

Dep. Variable:	Price	**R-squared:**	0.938
Model:	OLS	**Adj. R-squared:**	0.937
Method:	Least Squares	**F-statistic:**	931.6
Date:	Mon, 14 Mar 2022	**Prob (F-statistic):**	4.69e-148
Time:	17:18:57	**Log-Likelihood:**	-1439.8
No. Observations:	253	**AIC:**	2890.
Df Residuals:	248	**BIC:**	2907.
Df Model:	4		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
Intercept	2225.6083	45.448	48.971	0.000	2136.096	2315.121
Day	24.3821	3.432	7.105	0.000	17.623	31.141
np.sqrt(Day)	-199.9354	63.822	-3.133	0.002	-325.637	-74.234
np.square(Day)	-0.0481	0.004	-11.574	0.000	-0.056	-0.040
np.log(Day)	100.0696	69.986	1.430	0.154	-37.773	237.912

Let us call this model Model 3.

The log term is not statistically significant but R^2 improved!

Regression for Time Series: Crucial Questions

- Crucial Question 1: Is there a relationship between the response and the predictor.
- We test the null hypothesis:

$$H_0 : \beta_1 = \beta_2 = ... = \beta_n = 0$$

versus the alternative hypothesis

$$H_1 : \text{at least one } \beta_k \text{ is non-zero}$$

- This is computed through the F-statistic. We typically check for the p-value of the F-statistic being less than 5%.

Regression for Time Series: Crucial Questions

- Crucial Question 1b: Is there a relationship between a particular response and the predictor.
- We test the null hypothesis:

$$H_0 : \beta_k = 0$$

versus the alternative hypothesis

$$H_1 : \beta_k \neq 0$$

- This is computed through the t-statistic. We typically check for the p-value of the t-statistic being less than 5%.

Regression for Time Series: Crucial Questions

- Crucial Question 2: Which variables are important?
- Assume that the F-test has a small p-value and the regression is significant. Which predictors are important?
- One answer would be to fit all models that include subsets of all potential predictors. For each model, we can then check adjusted R^2, AIC, BIC and other similar criteria. Unfortunately, when there are n potential predictors, the total number of subsets is 2^n. This is impractical when n is large.
- We'll spend some time later on this critical question.

Regression for Time Series: Crucial Questions - Example

- Recall that we fitted three different models to the Google share price data (all data points), Model 1 (one predictor), Model 2 (three predictors), Model 3 (four predictors). Model 3 has the highest R^2 followed by Model 2 and Model 1 last.
- It turns out that we can always improve R^2 and MSE by adding more terms.
  - This comes from the least squares optimization formulation. If we have more degrees of freedom, we can always improve the objective function.

Regression for Time Series: Crucial Questions - Example

- To avoid model overfitting, the appropriate approach for model selection is to separate the training and test samples
  - Let's take the first 180 days as the training data
  - And the remaining 73 days as the test data
- We fit the models on the training data and check its error performance on the test data.

Regression for Time Series: Crucial Questions - Example

- Here are the results:

Model	Test Set RMSE
Model 1	415.68
Model 2	105.02
Model 3	116.42

- Model 3 has more predictors but performs worse than Model 2 on the test set. This is a sign of overfitting.
- Based on this particular train-test split, we'll prefer Model 2
  - But we can run more validation tests with different train-test splits.

Regression for Time Series: Crucial Questions

- Crucial Question 3: How strong is the model fit?
- We check R^2 and RMSE (or its corrected version the Residual Standard Error (RSE))

$$\text{RSE} = \sqrt{\frac{\text{RSS}}{T - p - 1}}$$

where

$$\text{RSS} = \sum_{t=1}^{T} (y_t - \hat{y}_t)^2$$

- Plot the data and the predictions

Regression for Time Series: Crucial Questions

- Crucial Question 4: Predictions
- The regression gives the mean of the predicted response but using the RSE we can compute confidence intervals.
- A $(1-\alpha)\%$ confidence interval is expected to contain the true observation $(1-\alpha)\%$ of the time.

Regression for Time Series: Basic Predictors: Categorical Variables

Other useful tricks for time series:

- We can use dummy variables to incorporate the effects of non-numerical (categorical predictors). For instance, assume that we are trying to predict daily demand but notice that weekends are considerably different than weekdays. We can then use a binary variable x_{wt} that takes the value of 1 if day t corresponds to a weekend day.

$$y_t = \beta_0 + \beta_1 x_{wt} + \epsilon_t$$

Note that our prediction would be $\hat{y}_t = \hat{\beta}_0$ for a weekday and $\hat{y}_t = \hat{\beta}_0 + \hat{\beta}_1$ for a weekend day.

Note that we only need one predictor for a binary variable (weekend vs. weekday). If it's not a weekend then it must be a week day.

Regression for Time Series: Basic Predictors: Categorical Variables

- Dummies can be used to model multiple categories. We can distinguish six dummies to mark the different days of the week. We don't need the seventh one since it would be dependent on the other six. x_{1t} is equal to 1 if it's a Monday and is 0 otherwise, x_{2t} is equal to 1 if it's a Tuesday and is 0 otherwise,..., x_{6t} is equal to 1 if it's a Saturday and is 0 otherwise, The model is then:

  $$y_t = \beta_0 + \beta_1 x_{1t} + \beta_2 x_{2t} + \ldots + \beta_6 x_{6t} + \epsilon_t$$

  Note that our prediction for a Monday would be $\hat{y}_t = \hat{\beta}_0 + \hat{\beta}_1$. For a sunday it would be $\hat{y}_t = \hat{\beta}_0$.

Regression for Time Series: Basic Predictors: Categorical Variables

- We can use dummies to mark months of the year, quarters of the year, hours of the day etc.
- We can also use dummies to mark irregular (non-seasonal) exceptions (holidays, days of Ramadan, promotions, school holidays etc.)
- This is great but note that we may easily end up with a very large number of dummies!

Regression for Time Series: Basic Predictors: Australian Beer Production

- The Australian Beer Production Data is strongly seasonal. We can try to fit:

$$y_t = \beta_0 + \beta_1 t + \beta_2 x_{1t} + \beta_3 x_{2t} + \ldots + \beta_{12} x_{11,t} + \epsilon_t$$

- where $x_{1t}, \ldots, x_{11,t}$ are the monthly dummies.

In [16]:
```
df = pd.read_csv('ausbeer_dummies.csv', index_col=0)
df.head()
```

Out[16]:

Month	Production	t	M1	M2	M3	M4	M5	M6	M7	M8	M9	M10	M11
1	164	1	1	0	0	0	0	0	0	0	0	0	0
2	148	2	0	1	0	0	0	0	0	0	0	0	0
3	152	3	0	0	1	0	0	0	0	0	0	0	0
4	144	4	0	0	0	1	0	0	0	0	0	0	0
5	155	5	0	0	0	0	1	0	0	0	0	0	0

Regression for Time Series: Basic Predictors: Australian Beer Production

$$y_t = \beta_0 + \beta_1 t + \beta_2 x_{1t} + \beta_3 x_{2t} + \ldots + \beta_{12} x_{11,t} + \epsilon_t$$

Dep. Variable:	Production	**R-squared:**	0.836
Model:	OLS	**Adj. R-squared:**	0.791
Method:	Least Squares	**F-statistic:**	18.30
Date:	Tue, 08 Mar 2022	**Prob (F-statistic):**	3.97e-13
Time:	12:20:51	**Log-Likelihood:**	-194.93
No. Observations:	56	**AIC:**	415.9
Df Residuals:	43	**BIC:**	442.2
Df Model:	12		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
Intercept	192.9750	5.015	38.477	0.000	182.861	203.089
t	-0.2158	0.075	-2.887	0.006	-0.367	-0.065
M1	-39.7792	6.030	-6.597	0.000	-51.940	-27.619
M2	-48.5633	6.026	-8.059	0.000	-60.715	-36.411
M3	-30.9475	6.023	-5.139	0.000	-43.093	-18.802
M4	-46.7317	6.020	-7.762	0.000	-58.873	-34.591
M5	-46.1158	6.019	-7.662	0.000	-58.254	-33.978
M6	-58.5000	6.018	-9.720	0.000	-70.637	-46.363
M7	-51.8842	6.019	-8.620	0.000	-64.022	-39.746
M8	-42.2683	6.020	-7.021	0.000	-54.409	-30.127
M9	-46.6475	6.348	-7.349	0.000	-59.449	-33.846
M10	-19.6817	6.346	-3.102	0.003	-32.479	-6.884
M11	-2.9658	6.344	-0.467	0.643	-15.760	9.829

Regression for Time Series: Basic Predictors: Australian Beer Production

$$y_t = \beta_0 + \beta_1 t + \beta_2 x_{1t} + \beta_3 x_{2t} + \ldots + \beta_{12} x_{11,t} + \epsilon_t$$

	coef	std err	t	P>|t|	[0.025	0.975]
Intercept	192.9750	5.015	38.477	0.000	182.861	203.089
t	-0.2158	0.075	-2.887	0.006	-0.367	-0.065
M1	-39.7792	6.030	-6.597	0.000	-51.940	-27.619
M2	-48.5633	6.026	-8.059	0.000	-60.715	-36.411
M3	-30.9475	6.023	-5.139	0.000	-43.093	-18.802
M4	-46.7317	6.020	-7.762	0.000	-58.873	-34.591
M5	-46.1158	6.019	-7.662	0.000	-58.254	-33.978
M6	-58.5000	6.018	-9.720	0.000	-70.637	-46.363
M7	-51.8842	6.019	-8.620	0.000	-64.022	-39.746
M8	-42.2683	6.020	-7.021	0.000	-54.409	-30.127
M9	-46.6475	6.348	-7.349	0.000	-59.449	-33.846
M10	-19.6817	6.346	-3.102	0.003	-32.479	-6.884
M11	-2.9658	6.344	-0.467	0.643	-15.760	9.829

Our prediction for month 4 is: 192.98 -0.2158 (4) -46.73

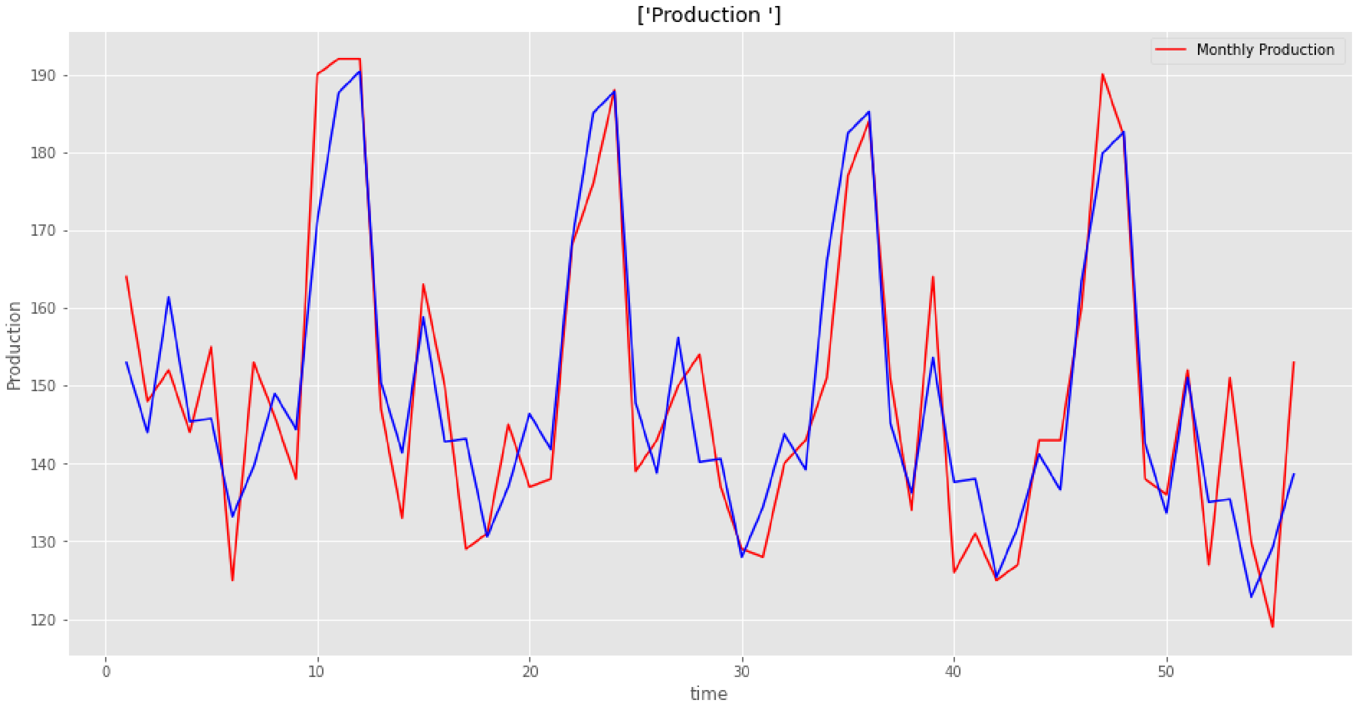
Our prediction for month 11 is: 192.98 -0.2158 (11) – 2.97

Our prediction for month 12 is: 192.98 -0.2158 (12)

Our prediction for month 26 is: 192.98 -0.2158 (2) -48.56

Month 12 is clearly the peak month for sales, all other months have negative seasonality factors wrt to month 12.

Regression for Time Series: Basic Predictors: Australian Beer Production



In-sample predictions in blue, and the observed production in red.

```
In [9]: error_beer = prod - result_beer.fittedvalues
```

```
In [10]: mse_beer = np.mean(np.square(error_beer ))
         rmse_beer = np.sqrt(mse_beer)
         mae_beer = np.mean(np.abs(error_beer ))
         mape_beer = np.mean(np.abs(error_beer )/prod)
         print('MSE Beer = ', mse_beer)
         print('RMSE Beer = ', rmse_beer)
         print('MAE Beer = ', mae_beer)
         print('MAPE Beer = ', mape_beer)

         MSE Beer =  61.805178571428556
         RMSE Beer =  7.861626967201417
         MAE Beer =  6.441249999999998
         MAPE Beer =  0.04378739163608506
```

Regression for Time Series: Basic Predictors: Categorical Variables

- We can use dummies to mark months of the year, quarters of the year, hours of the day etc.
- We can also use dummies to mark irregular (non-seasonal) exceptions (holidays, days of Ramadan, promotions, school holidays etc.)
- This is great but note that we may easily end up with a very large number of dummies!