TECHNICAL UNIVERSITY OF DENMARK

02435 DECISION-MAKING UNDER UNCERTAINTY

Assignment 2

Jorge Montalvo Arvizu, s192184

Spring 2020

Introduction

In this assignment, the overall problem setting is a stakeholder that wants to invest in real estate in four different areas: Zip2000, Zip2800, Zip7400, and Zip8900 in Denmark. The historic data of prices per m^2 for the four areas is given and the following analysis are used to help the investor to make decisions on the real estate assets: Linear Time Series Analysis and Stochastic Programming.

Task 1 - Linear Time Series Analysis

Before fitting the model, it is important to see if any transformation or differentiation is necessary. Since, in order to create time series model, the data should be stationary, i.e. mean and variance not changing over time and without increasing or decreasing trends.

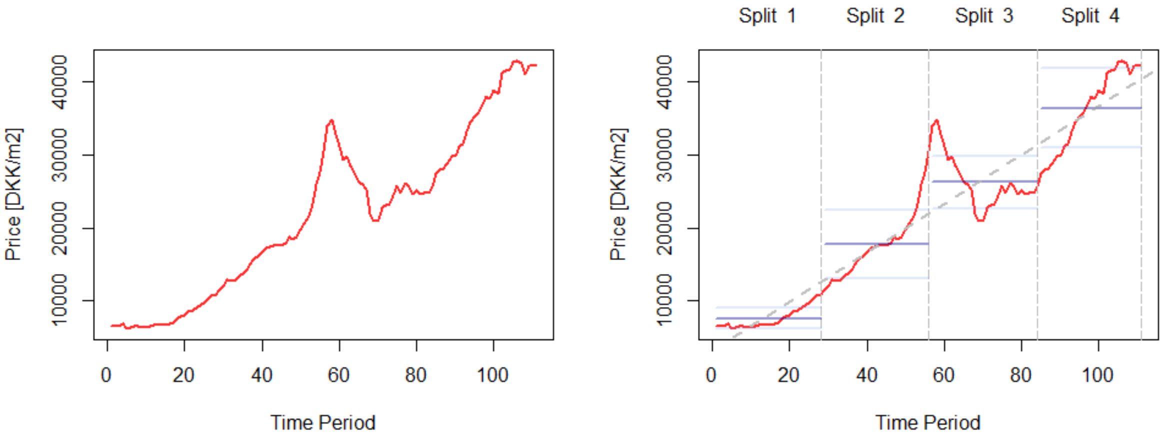


Figure 1: Price per m^2 in area Zip2000. On the left, the raw data plotted. On the right, the raw data plotted with four splits showing the variance of each split in dark transparent blue and the square root of the variance (i.e. standard deviation) in light transparent blue.

Split	Mean	Variance	Std. Deviation
1	7553	2123096	1457
2	17758	22154704	4706
3	26262	13033461	3610
4	36460	29558286	5436

Table 1: Descriptive statistics per split

It can be seen from Figure 1 and Table 1 that the mean and variance of the splits are different and changing over time, and there is an upwards trend in light gray, i.e. the data is non-stationary. To fix this non-stationarity, it is necessary to differentiate the raw data to remove the increasing trend and apply the log transformation to stabilize the variance. From Figure 2 on the right, it can be seen that the time series looks better when differentiating the data but there's still some trend of increasing variance over time; between time period 0 and 40 the variance behaves in certain range while from time period 40 to 110 the variance seems bigger. This behaviour can be corrected by log transforming the data. After the differentiation and the log transformation, it can be seen from Figure 3 on the right, that the data looks stationary.

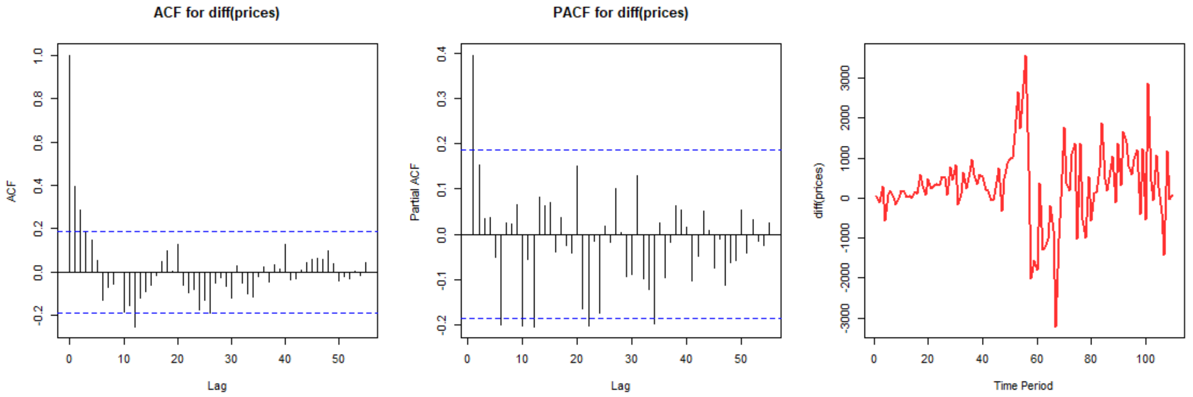


Figure 2: ACF, PACF and data plots with data differentiated.

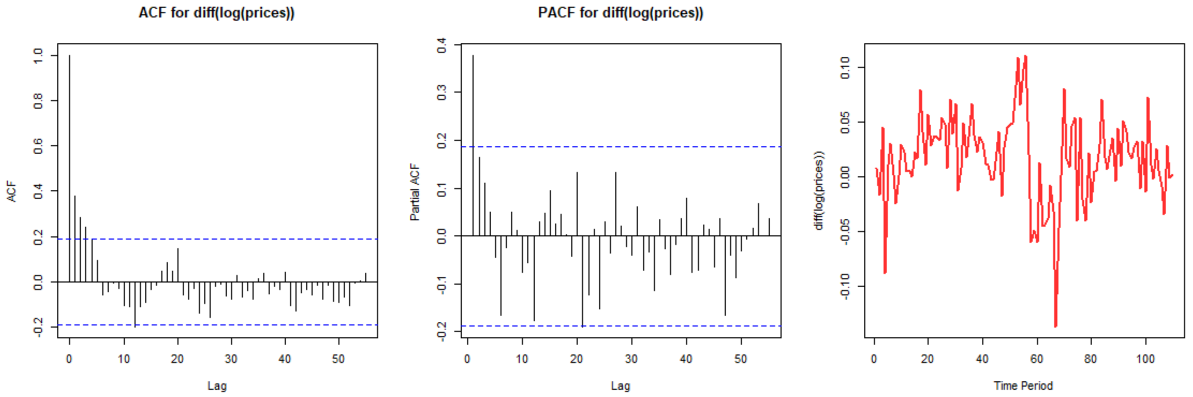


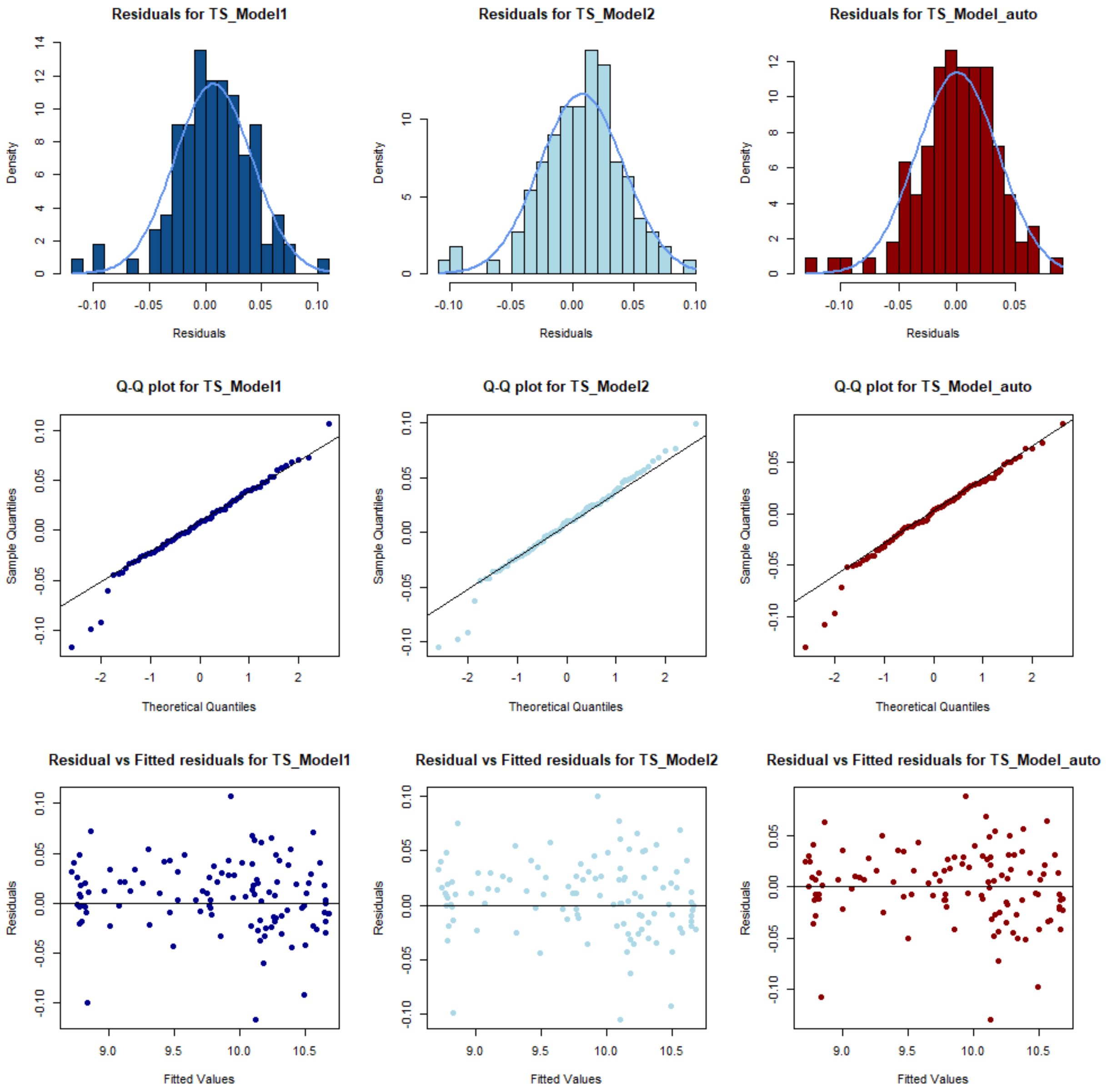
Figure 3: ACF, PACF and data plots with data differentiated and log transformed.

From the ACF and PACF plots from Figure 3, the p (from PACF) and q (from ACF) values can be extracted. In this case $q = 1$ since the PACF plot shows only one lag at 1 outside the statistical significance region; while $p = 4$ or $p = 5$ since the lags decrease from 0 to 4 or 5 outside the statistical significance region and then fluctuates inside it. The auto.arima function is also tested with the hand-made models resulting in the following ARIMA models:

Model	p	d	q
TS_Model1	1	1	4
TS_Model2	1	1	5
TS_Model_auto	1	1	1+drift*

Table 2: ARIMA models. Note that the auto.arima function created the TS_Model_auto and uses q as 1 but includes a drift parameter to account for the increasing trend.

These models are then evaluated given the distribution of the residuals, i.e. they should behave as a normal distribution. It can be seen from Figure 4 that the three models are quite similar. However, we can see a small (almost unnoticeable) shift to the right on the residuals plot for TS_Model2 and TS_Model_auto. Then, the line from the qqplot of TS_Model2 is slightly off from the 1st to the 2nd theoretical quantile. Finally, the residual vs. fitted residuals of all models are okay, i.e. like stars in the sky.



Residuals for TS_Model1

Residuals for TS_Model2

Residuals for TS_Model_auto

Q-Q plot for TS_Model1

Q-Q plot for TS_Model2

Q-Q plot for TS_Model_auto

Residual vs Fitted residuals for TS_Model1

Residual vs Fitted residuals for TS_Model2

Residual vs Fitted residuals for TS_Model_auto

Figure 4: Residuals, QQ-plot and residuals vs. fitted plots from the three models: on the left with dark blue TS_Model1, on the center with light blue TS_Model2 and on the right with dark red TS_Model_auto.

Given these diagnostics and the fact that the AIC of the three models were basically the same value at -410, the selected model is TS_Model1, which is an ARIMA(1,1,4) model with the following formula:

$$(1-\sum_{j=1}^{p}\phi_j B^j)(1-B)^d y_t = (1-\sum_{j=1}^{q}\theta_j B^j)e_t \tag{1}$$

If $p = 1$, $d = 1$, and $q = 4$:

$$(1-\phi_1 B)(1-B)y_t = (1-\sum_{q=1}^{4}\theta_j B^j)e_t \tag{2}$$

Then, with coefficients $\phi_1 = 0.439$, $\theta_1 = -0.1217$, $\theta_2 = 0.1569$, $\theta_3 = 0.1687$, and $\theta_4 = 0.1670$ and a white noise error term e_t:

$$(1-0.439B)(1-B)y_t = (1+0.1217B-0.1569B^2-0.1687B^3-0.1670B^4)e_t \tag{3}$$

With this model, the forecast for 1 step ahead is shown in Figure 5 with a value of 42469.03 DKK/m2 and 95% confidence intervals at the range between 40991.77 DKK/m2 and 43999.52 DKK/m2.

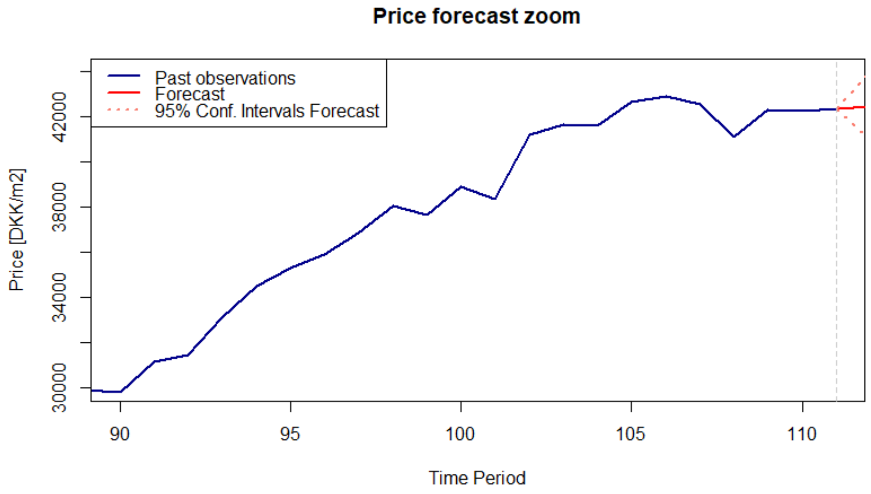


Figure 5: Price forecast (zoomed) for one time-step with 95% confidence intervals.

Task II - Scenario Generation

Using model TS_Model1 (3), 100 scenarios are now generated for the time series model for area Zip2000 using the simulation function for $\Omega = 100$. These can be seen on Figure 6, it can be seen that the lines are well dispersed for the forecasted time step 112. Given that it is not computationally possible to simulate all the possible realizations of the random variable price, this approach of scenarios generation is used to approximate the continuous process by a discrete stochastic process.

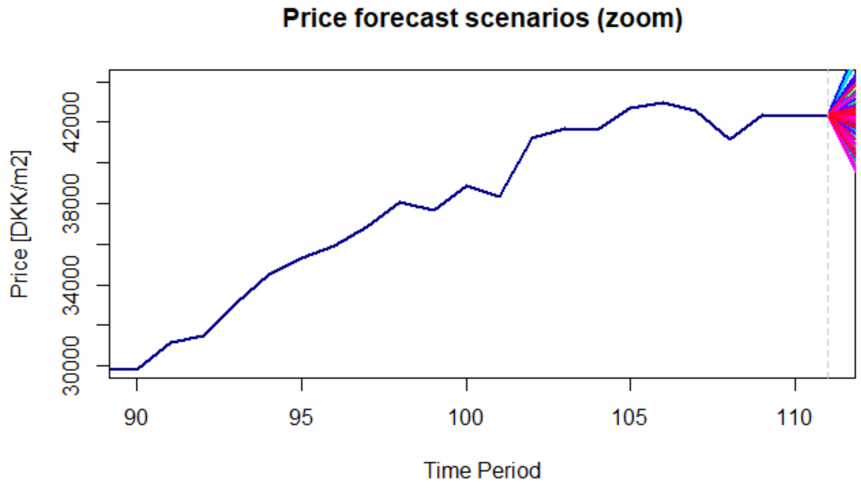


Figure 6: Forecast of 100 simulated scenarios.

Then, to further reduce the scenarios and select 10 representative scenarios by clustering scenarios with high similarity, the partitioning around medoids (PAM) method is used. These Ω scenarios are clustered into reduced scenarios $\omega = 10$ and can be seen on Figure 7, where the point forecasts represent the centroid/medoid results of the clustering.

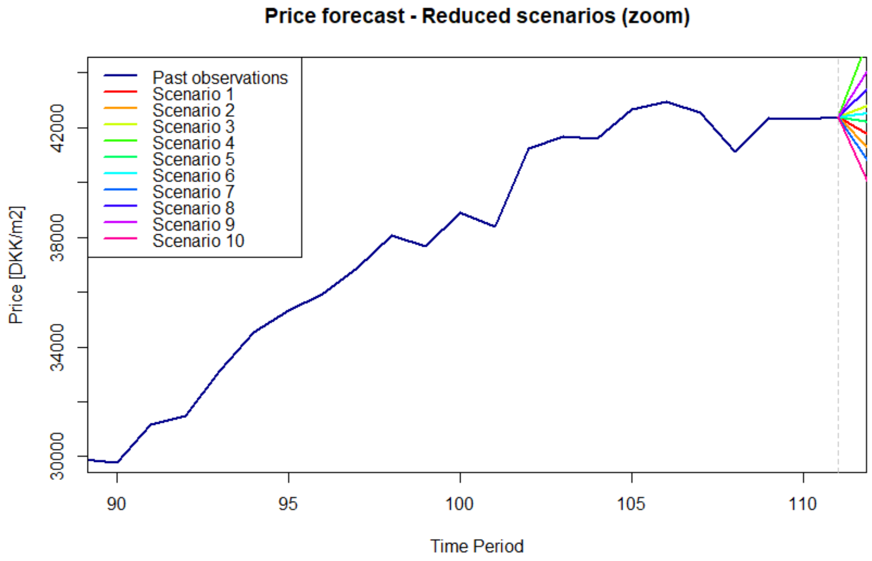


Figure 7: Forecast of 10 simulated scenarios from the PAM method.

Finally, the probability of each scenario is calculated as the sum of probabilities of scenarios in the cluster, thus giving the results in Table 3 where scenario 5 contains the highest probability and scenario 4 the lowest; it can be seen that scenario 5 is a point estimate close to the price forecast from the ARIMA model, while scenario 4 and 10 are at the extremes.

Scenario	1	2	3	4	5	6	7	8	9	10
Probability	0.11	0.11	0.12	0.05	0.13	0.11	0.09	0.12	0.09	0.07

Table 3: Scenarios probabilities

Task III - Stochastic Programming Formulation

1. General two-stage stochastic program:

Sets	
$\mathcal{I}$	Set of areas {Zip2000, Zip2800, Zip7400, Zip8900}
$\mathcal{S}$	Set of scenarios {s = 1 to 12}
Parameters	
p_i^{INIT}	Initial price for buying one m^2 in area i [DKK/m^2]
$P_{i,s}$	Future price forecast per scenario s and area i [DKK/m^2]
π_S	Probability per scenario s [-]
α	Confidence level for CVaR [-]
β	Weight factor [-]
B	Initial budget [DKK]
Free variables	
η	VaR, i.e. 1-α quantile of the distribution [-]
Positive variables	
x_i	Area bought (and held) at zip area i [m^2]
δ_s	Positive deviations between η and objective per scenario s [-]

Mathematical model:

$$max \quad (1-\beta)\{\sum_{s\in\mathcal{S}}\sum_{i\in\mathcal{I}}\pi_s p_{i,s}x_i - \sum_{i\in\mathcal{I}} p_i^{\text{INIT}}x_i\} + \beta\{\eta - \frac{1}{1-\alpha}\sum_{s\in\mathcal{S}}\pi_s\delta_s\} \tag{4}$$

$$s.t. \quad \sum_{i\in\mathcal{I}} p_i^{\text{INIT}}x_i = B \tag{5}$$

$$\eta - \{\sum_{i\in\mathcal{I}} {}_{i,s}x_i - \sum_{i\in\mathcal{I}} p_i^{\text{INIT}}x_i\} \leq \delta_s, \qquad \forall s \in \mathcal{S} \tag{6}$$

The objective function (4) maximizes the gain (where the value of the assets in the future minus the investment is weighted by (1-β)) and takes into account risk management with CVaR to improve the outcome in the not so profitable scenarios and is weighted by β. Constraint (5) enforces the budget to be spent fully. Finally, constraint (6) selects the scenarios that have a value $\leq$ to η. In this problem, first-stage variable is x_i, while second stage variables are η and δ_s, i.e. the variables related to risk management with CVaR. The results are the following:

```
Gain: 945708.827
CVaR: -4778259.362
Objective: -199084.811
Area zip2000: 3.0 m²
Area zip2800: 326.0 m²
Area zip7400: 13.0 m²
Area zip8900: 981.0 m²
```

It can be seen that the financial gain is positive, while CVaR and the objective function is negative. It is also interesting that the areas bought at areas zip2000 and zip7400 are just 3 and 13 m^2 respectively. These are really small areas for an investor to buy, however, the model obtains this solution as the most profitable while taking into account the CVaR at a confidence level α of 0.9.

Now, the mathematical formulation of the expected value version of the stochastic program is the following:

2. Expected value stochastic program:

Sets	
$\mathcal{I}$	Set of areas {Zip2000, Zip2800, Zip7400, Zip8900}
$\mathcal{S}$	Set of scenarios {s = 1 to 12}
Parameters	
p_i^{INIT}	Initial price for buying one m^2 in area i [DKK/m^2]
$Pexpected_i$	Expected price per area i [DKK/m^2]
B	Initial budget [DKK]
Positive variables	
x_i	Area bought (and held) at zip area i [m^2]

Notice that the CVaR parameters are not part of the model anymore and the first stage variable x_i is fixed after the first run of the stochastic program (without CVaR), hence, it is the same value as if only running the 1-stage stochastic program. The parameter $Pexpected_i$ is calculated as follows to obtain the expected value of the price at each zip area:

$$\mathbb{E}[p_{i,s}] = Pexpected_i = \sum_{s\in\mathcal{S}}\pi_s p_{i,s}, \qquad \forall i \in \mathcal{I} \tag{7}$$

Therefore, the mathematical model is:

$$max \quad \sum_{i \in \mathcal{I}} Pexpected_i x_i - \sum_{i \in \mathcal{I}} p_i^{\text{INIT}} x_i \tag{8}$$

$$s.t. \quad \sum_{i \in \mathcal{I}} p_i^{\text{INIT}} x_i = B \tag{9}$$

The objective function (8) maximizes the gain, i.e. the value of the assets in the future minus the investment. Constraint (9) enforces the budget to be spent fully. The results are the following:

```
Gain: 1166467.159
Objective: 1166467.159
Area zip2000: 8.0 m²
Area zip2800: 724.0 m²
Area zip7400: 28.0 m²
Area zip8900: 25.0 m²
```

Model	Gain
	[DKK]
Expected Value	1166467
Stochastic w/CVaR	945708

Table 4: Comparison of financial gain between RP w/CVaR and EEV models

From the results, we can see that the Expected Value solution results have a bigger financial gain compared to the stochastic solution. This may be due to the fact that the stochastic program is also optimizing for the CVaR and may affect the financial gain since it avoids the worst cases. It is important to check these models and its assumptions with an out-of-sample test that follows.

Task IV - Out-of-sample test

With the given samples, the financial gain was calculated using only the financial gain part of formula 4 and 8, since the variables are now fixed/given. From Figure 8, it can be seen that the Expected Value solution has a more flatten curve with extreme values in both sides (positive and negative), the extreme negative value being the double compared to the extreme value negative in the stochastic solution. Since EEV uses the expected value of the price, it uses no information about uncertainty. On the other hand, the Stochastic with CVaR solution makes use of the uncertainty information through the scenarios and optimizes the variables taking into account the worst scenarios at a confidence interval α of 0.9. The steps needed to create the out-of-sample test were the following:

Algorithm 1 Out-of-sample test

```
Solve stochastic program and obtain first-stage solution $x^S$
Solve expected-value program and obtain first-stage solution $x^E$
for $s = 1, 2, \ldots, \mathcal{S}$ do
    Solve deterministic problem with fixed first-stage variables $x^S$
    Solve deterministic problem with fixed first-stage variables $x^E$
end for
Plot histograms
Compare
```

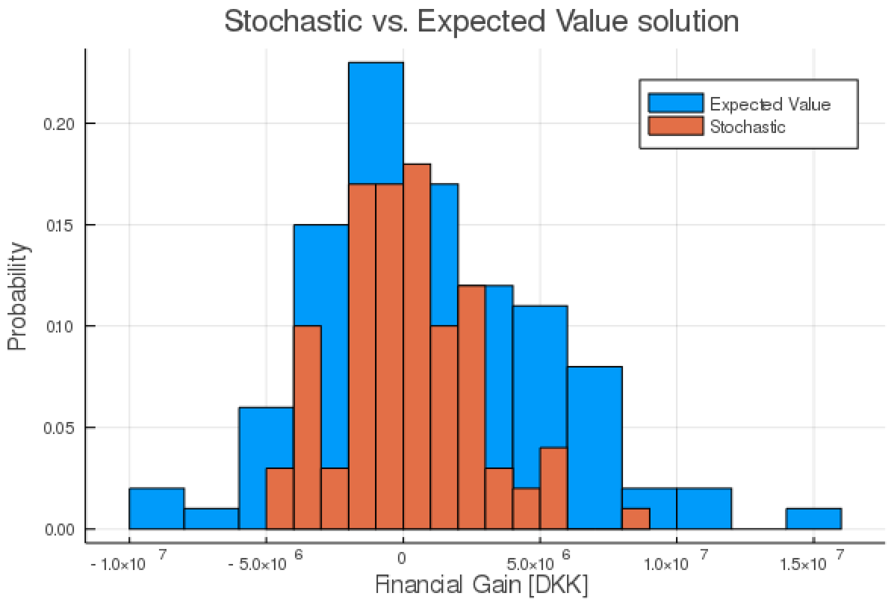


Figure 8: Histogram - Financial gain of EEV vs. Stochastic

It is important to use an out-of-sample test to assess for the robustness and applicability of the model since we're testing it against many "external" or "cleared uncertainty" scenarios/samples and therefore against scenarios that are possible real-world outcomes but aren't explicitly considered in each model. It is not enough to use EVPI and VSS because then it is just as comparing with the training data and not testing against real-world data. This way, we can compare the histograms of both models against the samples/scenarios and see which model behaves better for a risk-averse stakeholder. In this case, it is hard to select a model since both have quite some frequencies in the negative part of the histogram. However, the Expected Value solution contains more extreme values, which for a risk-averse agent that solution might expose him too much to risk and would prefer not to take chances, therefore the selected model would be the Stochastic with CVaR.

Conclusion

In conclusion, the investment decision during this assignment was tested with linear time series analysis and stochastic programming. In the first and second tasks, a forecast of the price in certain zip are was predicted once the data was made stationary and the model tested with normality diagnostics; then many scenarios were simulated and clustered into a few to assess the probability of each of them. Then two programs, one stochastic with CVaR and one expected value were modelled and used against the data available. These were later tested against real-world data and assessed for a risk-averse agent. It is important to notice that these solutions might change from agent to agent, since a riskier stakeholder might decrease the α of the Stochastic program to try to obtain a bigger financial gain while increasing its risk exposure. In this case, the decision was to avoid the extreme negative values and select the solution with a lower financial gain but more stable solutions.