Project Sources

* [https://www.analyticsvidhya.com/blog/2018/10/predicting-stock-price-machine-learningnd-deep-learning-techniques-python/?#](https://www.analyticsvidhya.com/blog/2018/10/predicting-stock-price-machine-learningnd-deep-learning-techniques-python/?)
* <https://www.analyticsvidhya.com/blog/2016/02/time-series-forecasting-codes-python/?utm_source=blog&utm_medium=stockmarketpredictionarticle>
* <https://www.analyticsvidhya.com/blog/2015/12/complete-tutorial-time-series-modeling/>
* <https://pandas.pydata.org/pandas-docs/stable/user_guide/computation.html>
* <https://towardsdatascience.com/predicting-stocks-with-machine-learning-1f8c53faaca3> - opinion about the approach
* <https://towardsdatascience.com/machine-learning-techniques-applied-to-stock-price-prediction-6c1994da8001> – Code follow implementation just for me
  1. <https://github.com/NGYB/Stocks/blob/master/StockPricePrediction/StockPricePrediction_v3_mov_avg.ipynb>
  2. <https://github.com/NGYB/Stocks/blob/master/StockPricePrediction/StockPricePrediction_v2_lin_reg.ipynb>
  3. <https://github.com/NGYB/Stocks/blob/master/StockPricePrediction/StockPricePrediction_v1c_xgboost.ipynb>
* <https://towardsdatascience.com/stock-price-prediction-intervals-using-monte-carlo-simulation-6b52b8ac9c27>
* <https://otexts.com/fpp2/stationarity.html> - for stationarity and models explanations
* <https://otexts.com/fpp2/planning.html> - forecasting explanation
* <https://otexts.com/fpp2/data-methods.html> - for the methods explanation
* <https://stats.stackexchange.com/questions/319604/linear-regression-independent-variable-stationarity> - stationarity in linear regression
* <https://machinelearningmastery.com/remove-trends-seasonality-difference-transform-python/> - again for stationarity. Good one
* <https://machinelearningmastery.com/time-series-data-stationary-python/> - stationarity – good
* <https://machinelearningmastery.com/difference-time-series-dataset-python/> - difference time series
* <https://machinelearningmastery.com/time-series-seasonality-with-python/> - remove seasonality
* <https://en.wikipedia.org/wiki/Dickey%E2%80%93Fuller_test> – formulas
* <https://www.statsmodels.org/stable/generated/statsmodels.tsa.stattools.adfuller.html> - Dickey-Fuller
* <https://towardsdatascience.com/analyzing-time-series-data-in-pandas-be3887fdd621>
* <https://people.duke.edu/~rnau/411diff.htm> – stationarity, Arima
* <https://en.wikipedia.org/wiki/Stationary_process> - definitions, formulas for stationarity
* <https://en.wikipedia.org/wiki/Seasonal_adjustment> – removing seasonality, definitions
* <https://en.wikipedia.org/wiki/White_noise> – definitions white noise
* <https://machinelearningmastery.com/white-noise-time-series-python/>
* Using Close price or Adj Close price

1. <https://quant.stackexchange.com/questions/40048/does-using-adjusted-closing-prices-constitute-a-lookahead-bias>
2. <https://quant.stackexchange.com/questions/24673/design-models-using-adjusted-or-unadjusted-stock-prices-time-series-prediction>
3. <https://quant.stackexchange.com/questions/25703/which-close-price-should-we-use-for-machine-learning>
4. <https://money.stackexchange.com/questions/21780/should-i-use-regular-or-adjusted-close-for-backtesting>
5. <https://budgeting.thenest.com/adjusted-closing-price-vs-closing-price-32457.html>
6. <https://www.sarpublisher.com/why-you-should-use-adjusted-closing-prices-for-analysis/>
7. <https://finance.zacks.com/adjusted-closing-price-vs-closing-price-9991.html>

* <https://stats.stackexchange.com/questions/103475/classifier-vs-model-vs-estimator>
* <https://alphascientist.com/data_management.html> - part of six articles for building a ML model
* <https://machinelearningmastery.com/vector-norms-machine-learning/> - for other project. Interesting info
* <https://github.com/Lucas-Kohorst/Python-Stock/blob/master/StocksStart.py>
* <https://www.red-gate.com/simple-talk/sql/bi/historical-stock-prices-volumes-python-csv-file/> - link for downloading data in csv
* <http://sebastianraschka.com/Articles/2015_pca_in_3_steps.html> – PCA
* <https://aroussi.com/post/python-yahoo-finance> – article about downloading data from Yahoo
* <https://jakevdp.github.io/PythonDataScienceHandbook/03.11-working-with-time-series.html>
* <https://towardsdatascience.com/basic-time-series-manipulation-with-pandas-4432afee64ea> - time series manipulations
* Train\_test\_split за времеви серии shuffle = False
* Explain technical and fundamental analysis – Check paper **Machine Learning Techniques for Stock Prediction – good paper for terms to explain**

**In theory, the Stock Prediction Problem can be considered as evaluating a function F at time T based on the previous values of F at times t-1, t-2, t-n while assigning corresponding weight function w at each point to F.**

* <https://quantivity.wordpress.com/2011/02/21/why-log-returns/> - explanation on why I will use log returns
* <https://en.wikipedia.org/wiki/Rate_of_return> - rate of return, info for compound ror
* <https://en.wikipedia.org/wiki/Compound_interest#Continuous_compounding>
* <https://www.portfolioprobe.com/2010/10/04/a-tale-of-two-returns/>
* For ARIMA model and for other useful things - Rapach, David, and Guofu Zhou, 2013, …
* NID - Normally and Independently Distributed (stochastic volatility – 91, Forecasting versus statistical hypothesis testing and estimation - 114 )
* <https://machinelearningmastery.com/arima-for-time-series-forecasting-with-python/> - ARIMA model
* <https://blog.quantinsti.com/forecasting-stock-returns-using-arima-model/> - ARIMA process!
* <https://blog.quantinsti.com/starting-time-series/> - a little bit about time series and stationarity
* <https://towardsdatascience.com/stock-market-analysis-using-arima-8731ded2447a> - малко графики, не е супер полезна
* <https://github.com/pierpaolo28/Kaggle-Challenges/blob/master/stock-market-analysis-and-time-series-prediction.ipynb> – малко работа с графики
* <https://www.datacamp.com/community/tutorials/tutorial-ridge-lasso-elastic-net> – good explanation for bias-variance and regularization
* <https://otexts.com/fpp2/expsmooth.html> – like an online textbook for working with time series, algorithms and so on!!!
* <https://people.duke.edu/~rnau/411diff.htm> – stationarity

<https://machinelearningmastery.com/remove-trends-seasonality-difference-transform-python/>

<https://machinelearningmastery.com/time-series-data-stationary-python/>

<https://machinelearningmastery.com/difference-time-series-dataset-python/>

<https://machinelearningmastery.com/time-series-seasonality-with-python/>

<https://machinelearningmastery.com/decompose-time-series-data-trend-seasonality/>

<https://machinelearningmastery.com/arima-for-time-series-forecasting-with-python/>

<https://machinelearningmastery.com/time-series-seasonality-with-python/>

<https://machinelearningmastery.com/seasonal-persistence-forecasting-python/>

<https://machinelearningmastery.com/time-series-data-visualization-with-python/>

<https://machinelearningmastery.com/time-series-trends-in-python/>

<https://people.duke.edu/~rnau/411diff.htm>

<https://www.dataquest.io/blog/tutorial-time-series-analysis-with-pandas/>

<https://towardsdatascience.com/analyzing-time-series-data-in-pandas-be3887fdd621>

<https://towardsdatascience.com/python-for-finance-stock-portfolio-analyses-6da4c3e61054>

<https://towardsdatascience.com/regularization-in-machine-learning-76441ddcf99a>

<https://towardsdatascience.com/stock-market-analysis-using-arima-8731ded2447a>

<https://towardsdatascience.com/whats-the-difference-between-linear-regression-lasso-ridge-and-elasticnet-8f997c60cf29>

<https://towardsdatascience.com/bias-variance-and-regularization-in-linear-regression-lasso-ridge-and-elastic-net-8bf81991d0c5>

<https://towardsdatascience.com/ridge-and-lasso-regression-a-complete-guide-with-python-scikit-learn-e20e34bcbf0b>

<https://towardsdatascience.com/in-12-minutes-stocks-analysis-with-pandas-and-scikit-learn-a8d8a7b50ee7>

<https://www.analyticsvidhya.com/blog/2016/03/practical-guide-principal-component-analysis-python/>

[https://www.analyticsvidhya.com/blog/2018/10/predicting-stock-price-machine-learningnd-deep-learning-techniques-python/?#](https://www.analyticsvidhya.com/blog/2018/10/predicting-stock-price-machine-learningnd-deep-learning-techniques-python/?)

<https://www.analyticsvidhya.com/blog/2017/06/a-comprehensive-guide-for-linear-ridge-and-lasso-regression/>

<https://www.analyticsvidhya.com/blog/2015/12/complete-tutorial-time-series-modeling/>

<https://www.analyticsvidhya.com/blog/2016/02/time-series-forecasting-codes-python/?utm_source=blog&utm_medium=stockmarketpredictionarticle>

<https://alphascientist.com/feature_engineering.html>

<https://alphascientist.com/feature_selection.html>

<https://alphascientist.com/walk_forward_model_building.html>

<https://alphascientist.com/model_evaluation>

<http://francescopochetti.com/stock-market-prediction-part-introduction/>

<http://francescopochetti.com/stock-market-prediction-part-ii-feature-generation/>

<http://francescopochetti.com/part-iii-scikit-classification-algorithms/>

<http://francescopochetti.com/part-iv-modelfeature-selection/>

<http://francescopochetti.com/part-iv-results-test-set/>

<http://francescopochetti.com/part-vii-backtest-portfolio-performance/>

<http://francescopochetti.com/pythonic-cross-validation-time-series-pandas-scikit-learn/>

<https://www.datacamp.com/community/tutorials/time-series-analysis-tutorial>

<https://www.datacamp.com/community/tutorials/finance-python-trading>

<https://www.datacamp.com/courses/intermediate-python-for-data-science>

<https://programmingforfinance.com/2017/11/calculating-the-shiller-pe-ratio-cape-ratio-pe10-ratio/>

<http://akuederle.com/matplotlib-zoomed-up-inset>

<https://otexts.com/fpp2/expsmooth.html>

<https://cambridgecoding.wordpress.com/2016/01/03/getting-started-with-regression-and-decision-trees/>

<https://blog.quantinsti.com/forecasting-stock-returns-using-arima-model/>

<https://blog.quantinsti.com/starting-time-series/>

<https://evanlariviereblog.wordpress.com/2018/03/15/machine-learning-predicting-stock-prices-decision-trees-and-random-forests/>

<https://evanlariviereblog.wordpress.com/2018/04/25/one-hot-encoding-and-binning/>

<https://rubikscode.net/2018/10/29/stock-price-prediction-using-hidden-markov-model/>

<https://www.sciencedirect.com/science/article/pii/S2405918818300060>

**Lasso, ridge, elastic net**

<http://www.sthda.com/english/articles/37-model-selection-essentials-in-r/153-penalized-regression-essentials-ridge-lasso-elastic-net/>

<https://www.datacamp.com/community/tutorials/tutorial-ridge-lasso-elastic-net>

<https://hackernoon.com/an-introduction-to-ridge-lasso-and-elastic-net-regression-cca60b4b934f>

<https://www.statisticshowto.datasciencecentral.com/lasso-regression/>

<http://shop.oreilly.com/product/9781783985104.do>

<https://stats.stackexchange.com/questions/84012/choosing-optimal-alpha-in-elastic-net-logistic-regression> - good comment

<http://docs.h2o.ai/h2o/latest-stable/h2o-docs/data-science/algo-params/alpha.html> - alpha acceptable values

<http://docs.h2o.ai/h2o/latest-stable/h2o-docs/data-science/algo-params/lambda.html> - lambda

<https://www.researchgate.net/profile/Federico_Andreis/publication/321106005_Shrinkage_methods_ridge_lasso_elastic_nets/links/5a0da3d7aca2729b1f4eeabb/Shrinkage-methods-ridge-lasso-elastic-nets.pdf> -presentation

<https://web.stanford.edu/~hastie/Papers/B67.2%20(2005)%20301-320%20Zou%20&%20Hastie.pdf> = paper Stanford

<https://www.analyticsvidhya.com/blog/2016/01/complete-tutorial-ridge-lasso-regression-python/> - for alpha

<https://medium.com/predict/feature-engineering-for-stock-price-movement-54669d4a7efc>

<https://dachxiu.chicagobooth.edu/download/ML.pdf>

<https://link.springer.com/content/pdf/10.1186%2Fs40854-019-0138-0.pdf>

<https://arxiv.org/pdf/1903.06478.pdf>

<https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3428095>

<https://papers.ssrn.com/sol3/papers.cfm?abstract_id=1549328>

<https://papers.ssrn.com/sol3/papers.cfm?abstract_id=1586656>

<https://web.stanford.edu/~hastie/TALKS/enet_talk.pdf>

<https://ncss-wpengine.netdna-ssl.com/wp-content/themes/ncss/pdf/Procedures/NCSS/Ridge_Regression.pdf>

<https://www.investopedia.com/articles/basics/04/100804.asp>

<https://www.investopedia.com/terms/e/eps.asp>

<https://www.investopedia.com/terms/p/price-earningsratio.asp>

<https://www.investopedia.com/investing/primary-and-secondary-markets/>

<https://www.investopedia.com/terms/t/treasurybill.asp>

<https://www.investopedia.com/terms/r/risk-freerate.asp>

<https://www.investopedia.com/terms/r/risk-freereturn.asp>

<https://www.investopedia.com/terms/e/excessreturn.asp>

<https://www.investopedia.com/terms/t/treasury-yield.asp>

<https://www.treasury.gov/resource-center/data-chart-center/interest-rates/Pages/TextView.aspx?data=billrates>

<https://www.investopedia.com/ask/answers/102714/how-do-you-calculate-beta-excel.asp>

<https://www.investopedia.com/ask/answers/070615/what-formula-calculating-beta.asp>

<https://corporatefinanceinstitute.com/resources/knowledge/finance/beta-coefficient/>

<https://www.google.com/search?q=stationary+gausian+comparison&oq=stationary+gausian+comparison&aqs=chrome..69i57.11538j0j7&sourceid=chrome&ie=UTF-8>

<https://financeformulas.net/Total-Stock-Return.html>

<https://quantivity.wordpress.com/2011/02/21/why-log-returns/>

<https://chateauxdebah.wordpress.com/2014/08/10/why-use-log-returns/>

<https://www.google.com/search?q=lagged+observations+meaning&oq=lagged+observation&aqs=chrome.2.0j69i57j0l3.9643j1j7&sourceid=chrome&ie=UTF-8>

<https://stattrek.com/statistics/dictionary.aspx?definition=z-score>

<https://www.google.com/search?q=efficient+market+hypothesis&oq=efficient+market+hypothesis&aqs=chrome..69i57j0l5.5260j0j7&sourceid=chrome&ie=UTF-8>

<https://www.google.com/search?sxsrf=ACYBGNQcoY-Etz6GYNT5xN8a99AU_0PP1A%3A1573196507947&ei=2xLFXerEOdD2gQa4_ZWABQ&q=stock+data+with+many+indicators+earnings+free+download&oq=stock+data+with+many+indicators+earnings+free+download&gs_l=psy-ab.3...1817369.1826741..1827086...5.3..0.238.3499.3j26j1......0....1..gws-wiz.......0i71j33i10j33i21.n1Huj4roF6Y&ved=0ahUKEwiqnpLYhdrlAhVQe8AKHbh-BVAQ4dUDCAs&uact=5>

<https://www.google.com/search?q=what+is+ols+estimator&oq=what+is+ols+&aqs=chrome.2.69i57j0l5.8610j0j7&sourceid=chrome&ie=UTF-8>

<https://github.com/techdisrupt/Stock-Market-Data-Extraction-Python/blob/master/GetEPSandPE.py>

<https://www.ryansmccoy.com/2019/01/28/download-eps-sales-estimates-using-python/>

<https://stackoverflow.com/questions/31287552/logarithmic-returns-in-pandas-dataframe>

<https://simfin.com/data/access/download>

<https://www.quandl.com/data/FRED/DTB3-3-Month-Treasury-Bill-Secondary-Market-Rate>

<https://www.statsmodels.org/dev/tsa.html>

<https://www.statsmodels.org/dev/generated/statsmodels.tsa.statespace.sarimax.SARIMAX.html?fbclid=IwAR2UyT3z2UXp9ppVqu6rPQuj_yjI0VYAw_JAvPO4qynM9Zn-NPBRODUfEtM>

<https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.ElasticNet.html>

<https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.Ridge.html>

<https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.Lasso.html>

<https://risk.edhec.edu/investment-management-python-and-machine-learning>

<https://www.crummy.com/software/BeautifulSoup/bs4/doc/>

<https://stackoverflow.com/questions/37286465/python-matplotlib-finance-get-the-quarterly-eps-of-a-stock>

<https://blog.quantinsti.com/stock-market-data-analysis-python/#fundamental>

<https://stackoverflow.com/questions/9740116/programmatic-access-to-detailed-historical-financial-data>

<https://finance.yahoo.com/news/comparing-p-e-eps-earnings-174000440.html?guccounter=1&guce_referrer=aHR0cHM6Ly93d3cuZ29vZ2xlLmNvbS8&guce_referrer_sig=AQAAAIRio5ekSuHJ_Db2FJCudx7MU2hSPXz8rbM_YkTuGnjCKMefRWASnPzH9VXWKJb9pUnE53e2dl6i0ycbt8DVdxKhBdjww7PY7WTQdyWyvmLLbWP1z9OWoYx8X7XsAEAua4GIHil3mnGSTPFKWAQrH2KnueCuaGNmeKroMtP30n54>

<https://investinganswers.com/dictionary/t/treasury-bill>

<https://ntguardian.wordpress.com/2016/09/19/introduction-stock-market-data-python-1/>

<https://seekingalpha.com/article/4234024-stocks-machine-learning> -interesting

**Dickey-Fuller**

<https://en.wikipedia.org/wiki/Dickey%E2%80%93Fuller_test>

<https://en.wikipedia.org/wiki/Augmented_Dickey%E2%80%93Fuller_test>

**Autocorrelation**

<https://newonlinecourses.science.psu.edu/stat462/node/188/> - very useful

<https://stackoverflow.com/questions/36038927/whats-the-difference-between-pandas-acf-and-statsmodel-acf>

<https://stats.stackexchange.com/questions/355922/how-to-interpret-autocorrelation-plot-for-differencing>

<https://stats.stackexchange.com/questions/234492/what-is-the-difference-between-period-cycle-and-seasonality>

<https://stats.stackexchange.com/questions/134487/analyse-acf-and-pacf-plots> - useful

<https://support.minitab.com/en-us/minitab/18/help-and-how-to/modeling-statistics/time-series/supporting-topics/diagnostic-checking/what-is-the-ljung-box-q-statistic/> - LBQ statistic – good for assessment autocorrelation and ARIMA possibly.

<https://www.statsmodels.org/dev/generated/statsmodels.tsa.stattools.acf.html>

<https://stats.stackexchange.com/questions/263366/interpreting-seasonality-in-acf-and-pacf-plots>

**Reference for holidays**

<https://www.timeanddate.com/calendar/?year=2015&country=1>

<https://www.vercalendario.info/en/event/united_states-holidays-3-april-2015.html>

**Why returns not prices**

<https://quant.stackexchange.com/questions/16481/why-do-we-usually-model-returns-and-not-prices/16484>

<https://en.wikipedia.org/wiki/Rate_of_return#Logarithmic_or_continuously_compounded_return>

**Explain settle price**

<https://investinganswers.com/dictionary/s/settlement-price>

**Shift to make supervised data series**

<https://machinelearningmastery.com/convert-time-series-supervised-learning-problem-python/>

<https://www.geeksforgeeks.org/python-pandas-series-shift/>

**USD to ECU – monthly data**

<https://fred.stlouisfed.org/series/EXUSEC?cid=280>

<https://stackoverflow.com/questions/29612705/converting-pandas-dataframe-from-monthly-to-daily>

<https://en.wikipedia.org/wiki/European_Currency_Unit#Value_determined_by_basket_of_currencies>

**plotting data**

<https://pandas.pydata.org/pandas-docs/stable/user_guide/visualization.html>

<https://stackoverflow.com/questions/47383588/using-a-matplotlib-button-to-alternate-switch-between-plots-i-created>

<https://plot.ly/python/time-series/>

<https://blog.dominodatalab.com/interactive-dashboards-in-jupyter/>

<https://www.datadoghq.com/blog/timeseries-metric-graphs-101/> - very useful for knowing what type of plot I need

[http://www.blackarbs.com/blog/advanced-time-series-plots-in-python/1/6/2017 - plot recessions.very](http://www.blackarbs.com/blog/advanced-time-series-plots-in-python/1/6/2017%20-%20plot%20recessions.very) important

**correlogram**

<https://en.wikipedia.org/wiki/Correlogram>

<https://en.wikipedia.org/wiki/Seasonal_subseries_plot>

**Very good tutorial for plotting**

<http://cican17.com/data-visualization-with-python/>

Black Monday

<https://en.wikipedia.org/wiki/Black_Monday_(1987)>

Dot-com bubble

<https://en.wikipedia.org/wiki/Dot-com_bubble>

Financial Crisis

<https://en.wikipedia.org/wiki/Financial_crisis_of_2007%E2%80%932008>

United States housing bubble and housing correction

<https://en.wikipedia.org/wiki/United_States_housing_bubble#Timeline>

<https://en.wikipedia.org/wiki/United_States_housing_market_correction>

<https://en.wikipedia.org/wiki/United_States_housing_market_correction#Major_downturn_and_subprime_mortgage_collapse,_2007>

Flash Crash – to remove this date

<https://en.wikipedia.org/wiki/2010_Flash_Crash>

Black Friday

<https://en.wikipedia.org/wiki/Friday_the_13th_mini-crash> - not removing, junk bonds sellout

**Python for trading**

<https://learndatasci.com/tutorials/python-finance-part-yahoo-finance-api-pandas-matplotlib/>

**seasonal adjustment – it explains the arima process too**

<https://en.wikipedia.org/wiki/Seasonal_adjustment>

<https://en.wikipedia.org/wiki/Seasonal_subseries_plot>

<https://www.joyofdata.de/blog/category/machine-learning/>

**regression vertical/horizontal lines explanation, alpha, beta of a stock and other interesting info**

<https://www.aistockcharts.com/linear_correlation_explained.htm>

<http://www.shodor.org/interactivate/discussions/CorrelationCoefficients/>

<http://web.pdx.edu/~newsomj/pa551/lectur17.htm> - simple regression explained

<http://www.shodor.org/interactivate/discussions/CorrelationCoefficients/> - r value

**OLS, Dickey Fuller stationarity. Important file**

<http://conference.scipy.org/proceedings/scipy2011/pdfs/statsmodels.pdf>

**pvalue**

<https://blog.minitab.com/blog/understanding-statistics/what-can-you-say-when-your-p-value-is-greater-than-005>

**Reccesions chart**

<https://fred.stlouisfed.org/series/JHDUSRGDPBR>

**Very good tutorials on exponential smoothing**

<https://supchains.com/article/exponential-smoothing-python/>

<https://supchains.com/statistical-models/>

<https://supchains.com/article/exponential-smoothing-with-trend-python/>

<https://supchains.com/article/holt-winters-forecast-with-multiplicative-seasonality-python/>

<https://supchains.com/article/holt-winters-forecast-with-additive-seasonality/>

<https://machinelearningmastery.com/exponential-smoothing-for-time-series-forecasting-in-python/>

<https://machinelearningmastery.com/how-to-grid-search-triple-exponential-smoothing-for-time-series-forecasting-in-python/>

<https://otexts.com/fpp2/ses.html>

<https://www.analyticsvidhya.com/blog/2018/02/time-series-forecasting-methods/>

<https://blogs.oracle.com/datascience/7-effective-methods-for-fitting-a-linear-model-in-python>

<https://blogs.oracle.com/datascience/decomposition-based-approaches-to-time-series-forecasting>

**additive vs multiplicative decomposition**

<https://stats.stackexchange.com/questions/105432/additive-vs-multiplicative-decomposition>

**good for time series analysis - check – KPSS and ADF tests. Check if time left**

<https://www.machinelearningplus.com/time-series/time-series-analysis-python/>

<https://www.ibm.com/support/knowledgecenter/en/SS3RA7_15.0.0/com.ibm.spss.modeler.help/timeseries_acf_pacf.htm>

<https://www.ibm.com/support/knowledgecenter/en/SS3RA7_15.0.0/com.ibm.spss.modeler.help/timeseries_transformations.htm>

<https://www.ibm.com/support/knowledgecenter/en/SS3RA7_15.0.0/com.ibm.spss.modeler.help/timeseries_arima_criteria.htm>

**intra-weekly detrending – freq for seasonal decompose**

<https://www.bundesbank.de/resource/blob/763892/0d1c33f19a204e2233a6fccc6e802487/mL/2018-10-17-dkp-41-data.pdf>

<https://stats.stackexchange.com/questions/185058/what-will-be-frequency-for-daily-observation-in-time-series-and-how-to-deal-with>

<https://stats.stackexchange.com/questions/144158/daily-time-series-analysis/144281#144281>

<https://stats.stackexchange.com/questions/58657/time-series-forecasting-with-daily-data-arima-with-regressor/58673#58673>

<https://stats.stackexchange.com/questions/144509/forecast-daily-data-with-weekly-and-monthly-seasonality-using-exponential-smooth/144569#144569>

**pvalue zero for Dickey**

<https://stats.stackexchange.com/questions/125864/zero-p-value-for-dickey-fuller-test>

<https://www.statsdirect.com/help/basics/p_values.htm>

**nan in seasonal decompose**

<https://github.com/statsmodels/statsmodels/issues/3872>

**RANSAC**

<http://www.cse.psu.edu/~rtc12/CSE486/lecture15.pdf>

<http://www.cse.yorku.ca/~kosta/CompVis_Notes/ransac.pdf>

<https://en.wikipedia.org/wiki/Random_sample_consensus>

**Autocorrelation**

<https://en.wikipedia.org/wiki/Partial_autocorrelation_function>

<https://en.wikipedia.org/wiki/Autocorrelation#Regression_analysis>

<https://en.wikipedia.org/wiki/Pearson_correlation_coefficient>

**Errors and residuals**

<https://en.wikipedia.org/wiki/Errors_and_residuals>

**Polynomial regression**

<https://en.m.wikipedia.org/wiki/Polynomial_regression>

<https://www.geeksforgeeks.org/python-implementation-of-polynomial-regression/>

<https://en.m.wikipedia.org/wiki/Linear_regression#Simple_and_multiple_linear_regression> – heteroscedasticity

<https://en.m.wikipedia.org/wiki/Ordinary_least_squares> - OLS

<https://en.m.wikipedia.org/wiki/Generalized_least_squares> - GLS

<https://en.m.wikipedia.org/wiki/Heteroscedasticity>

<https://en.m.wikipedia.org/wiki/Skewness>

<https://en.m.wikipedia.org/wiki/Least_squares>

<https://en.m.wikipedia.org/wiki/Vandermonde_matrix>

<https://en.m.wikipedia.org/wiki/Spline_(mathematics)>

**Gridsearch and linear regression**

<https://stats.stackexchange.com/questions/153131/gridsearchcv-regression-vs-linear-regression-vs-stats-model-ols>

**Useful**

<https://www.machinelearningplus.com/time-series/time-series-analysis-python/>

<https://stats.stackexchange.com/questions/153131/gridsearchcv-regression-vs-linear-regression-vs-stats-model-ols>

<https://www.ritchieng.com/machine-learning-evaluate-linear-regression-model/> - good

**Pickle**

<https://pythonprogramming.net/python-pickle-module-save-objects-serialization/>

**r squared**

<http://www.fairlynerdy.com/what-is-r-squared/>

<https://newonlinecourses.science.psu.edu/stat462/node/98/>

<https://stats.stackexchange.com/questions/183265/what-does-negative-r-squared-mean> - very good explanation

<https://stats.stackexchange.com/questions/29781/when-conducting-multiple-regression-when-should-you-center-your-predictor-varia> - why scaling

<https://nbviewer.jupyter.org/github/Yorko/mlcourse_open/blob/master/jupyter_english/topic09_time_series/topic9_part1_time_series_python.ipynb?flush_cache=true> – TRY with classes and grid search manual

<https://stackoverflow.com/questions/34990652/why-do-we-need-np-squeeze/34991288> - NP squeeze

**from joblib import Parallel, delayed – check these**

**p, q values**

<https://stackoverflow.com/questions/57472146/decide-p-q-values-based-on-acf-and-pacf>

<https://www.statsmodels.org/stable/generated/statsmodels.tsa.arima_model.ARIMA.html>

<https://stackoverflow.com/questions/54622279/why-acf-pacf-has-different-lags-range>

<https://stackoverflow.com/questions/57349032/decide-p-q-values-based-on-acf-and-pacf-graphs-and-identify-parameters-of-sarim>

<https://stats.stackexchange.com/questions/241806/arima-modelling-identifying-p-q-and-p-q-from-acf-and-pacf-plots>

<https://people.duke.edu/~rnau/411arim3.htm#plots> – good

<https://people.duke.edu/~rnau/Slides_on_ARIMA_models--Robert_Nau.pdf> - good presentation, explain with this

<https://www.sciencedirect.com/topics/computer-science/convergence-tolerance> – important too

<https://en.wikipedia.org/wiki/Akaike_information_criterion#How_to_use_AIC_in_practice> – AIC