Predicting Patronage

When creators are paid, they can create more amazing things.
– Patreon.com

Saurabh Gupta

Supervisor: Dr Brendon J. Brewer
Department of Statistics, University of Auckland
Masters Research (30 points)

November 16, 2018

Outline

Background

- Patreon.com is a new way to fund creators.
- Hosts 122 thousand creators, 4 million pledges and growing!
- Enables patrons to sign up for a small monthly donation to video, music creators, publishers, etc.
- However, patrons can change their donations at any time.
- Fitting so many time series is in itself a challenge.

Objectives

- Bayesian model to forecast Patrons for top creators
- Compare with classical time series methods
- Evaluate impact of social media metrics on Patrons

The Dataset

28 months of time series data on the number of Patrons and 5 social media metrics for top 103 creators in different categories.

- Split into 24 months of training, 4 months of test data.
- Sufficient to predict one quarter ahead a month in advance.
- Social media metrics include number of Facebook Likes, Twitter Followers, YouTube Subscribers, YouTube Videos and YouTube Views.
- Data measured on the 1st of each month.

Data Management

- Data sourced from Graphtreon API as csv files.
- Extracted 103 of top 500 creators with no missing values for the response variable `Patrons`.
- Reshaped into time series for each creator contained in an **R** list object.
- Analysed mostly using `lapply` family of functions.
- Fit the Bayesian models using `bash`, **R** and `Python` scripts.

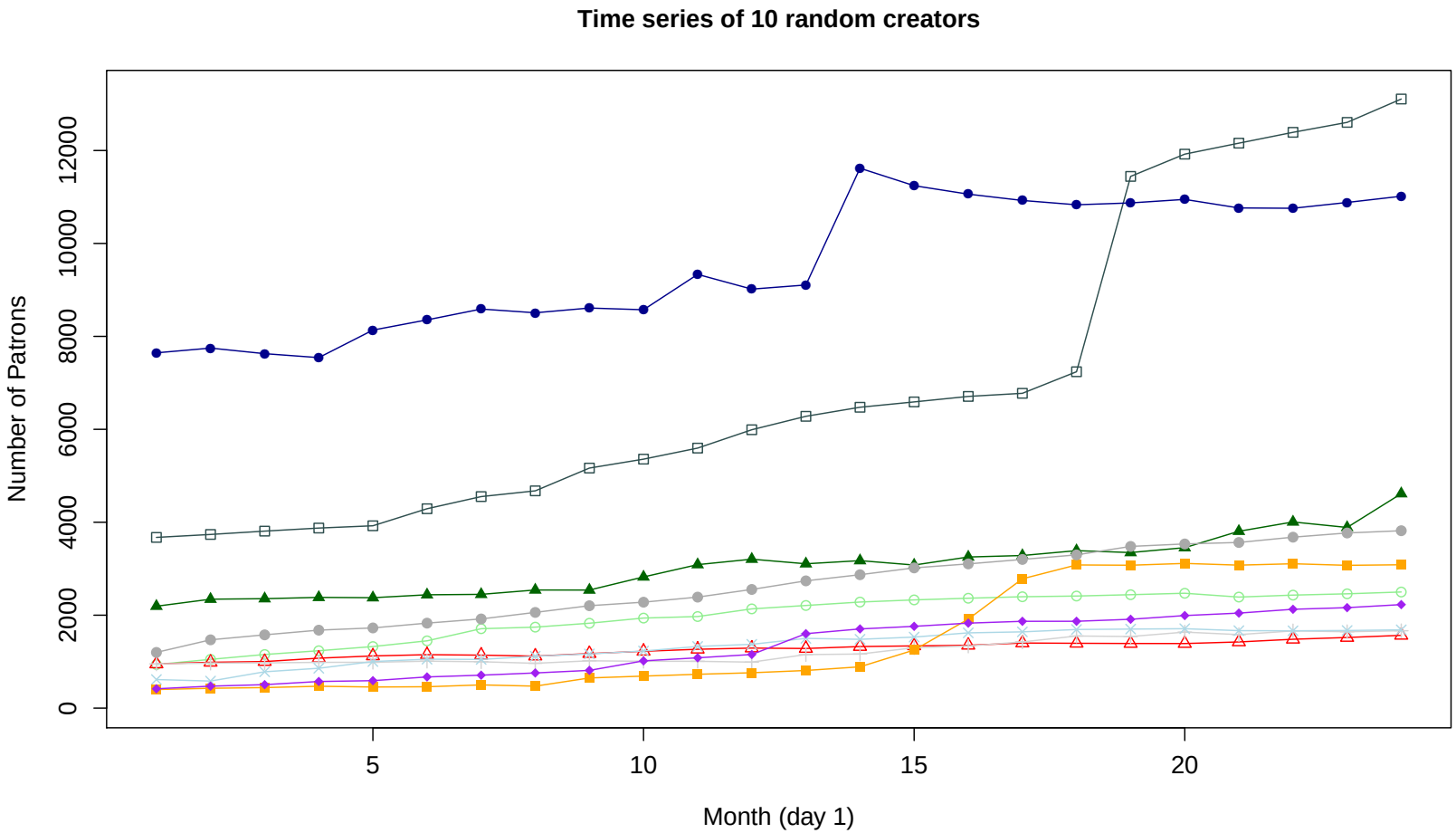
The Models

- Log Growth of Patrons
- Linear Trend model
- Quadratic Trend model
- For comparisons, two methods[1] were used for each model:
  - `auto.arima` from package `Forecast` for classical time series modelling[1].
  - Diffusive Nested Sampling for Bayesian models using `DNest4` in `C++` and `Python`.

[1] `auto.arima` automatically selects the model and its parameters based on statistical tests. Hence, in the results section, the same model output has been compared to the Bayesian linear and quadratic trend models.

Patrons of most creators have a linear trend

A few have one or two large shifts



Log Growth Model

Log monthly growth in `Patrons`, Y, defined as the response, $g_t \in G$:

$$g_t = \mu + \alpha(g_{t-1} - \mu) + w_t \tag{1}$$

$$\text{where, } g_t \equiv \ln(Y_t / Y_{t-1}) \tag{2}$$

$$\alpha \equiv \exp\left(-\frac{1}{L}\right) \tag{3}$$

Assumed distribution of the innovations, or White Noise:

$$w_t \overset{\text{iid}}{\sim} t(0, \beta, \nu) \tag{4}$$

Assumed prior distribution for the parameters:

$$\mu \sim t(0, 0.1, 2) \tag{5}$$

$$\ln L \sim \text{Uniform}(-10, 10) \tag{6}$$

$$\ln \beta \sim t(-2, 1, 2) \tag{7}$$

$$\ln \nu \sim \text{Uniform}(-1, 5) \tag{8}$$

Cross-correlation with log growth of other variables

Social media metrics are following Patrons for 10-20% creators.

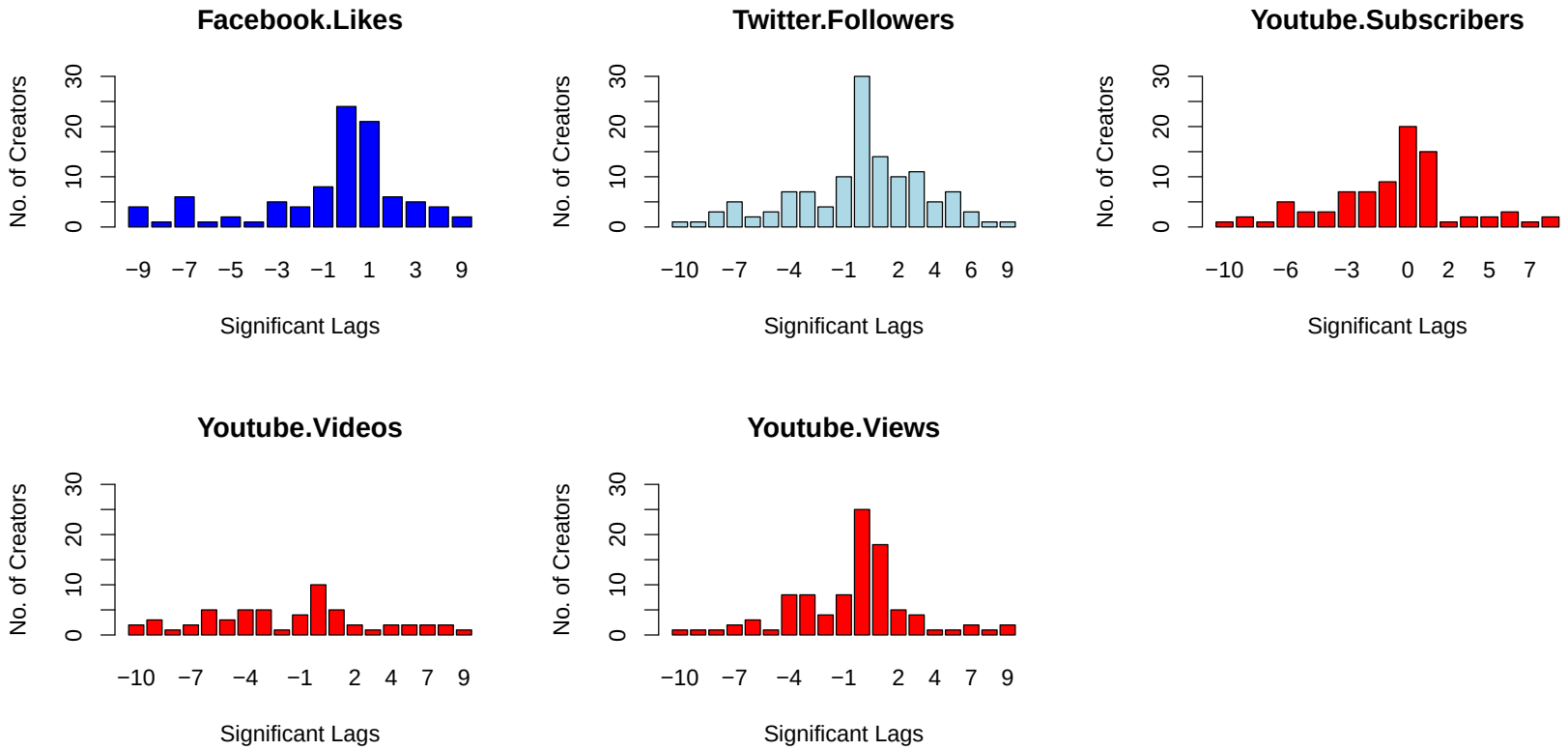


Figure: Cross-Correlation between log growth of variables. The plots are histograms of significant lags for 103 creators using `ccf` function in **R**. For our forecast, we need correlation for lag -4 or less.

Model with linear trend

Model for `Patrons`, Y, and an underlying AR(1) time series:

$$y_t = \beta_0 + \beta_1 t + \alpha(y_{t-1} - \beta_0 - \beta_1(t-1)) + \epsilon_t \tag{9}$$

where, Y has intercept (β_0) and slope (β_1) as a linear function of time period $T \in 1, \ldots, N$.

Assumed prior distribution for the parameters:

$$\beta_0 \sim \text{Uniform}(0, 10000) \tag{10}$$
$$\beta_1 \sim \text{Normal}(0, 100^2) \tag{11}$$
$$\log \sigma \sim \text{Uniform}(-5, 5) \tag{12}$$
$$\alpha \sim \text{Normal}(0, 1^2) \tag{13}$$

Significance of linear trend, and residual diagnostics

- Using **R** function `lm()` to test for linear trend, the coefficient for:
  - the intercept, β_0, was significant for 94 of 103 creators;
  - while the slope, β_1, was significant for 100 creators;
  - assuming a significance level of 0.05.
- The coefficients are significant even if Bonferroni correction is applied for multiple testing.
- Time series diagnostics such as ACF, PACF and CCF were conducted on the residuals after detrending the data.
- It indicated that if we fit an AR(1) model to the linear trend residuals of `Patrons`, we won't be over-fitting more than 20 creators. Hence, an AR(1) model was fit to the residuals.

Cross-correlation with other variables

Social media metrics can't help forecast four months ahead.

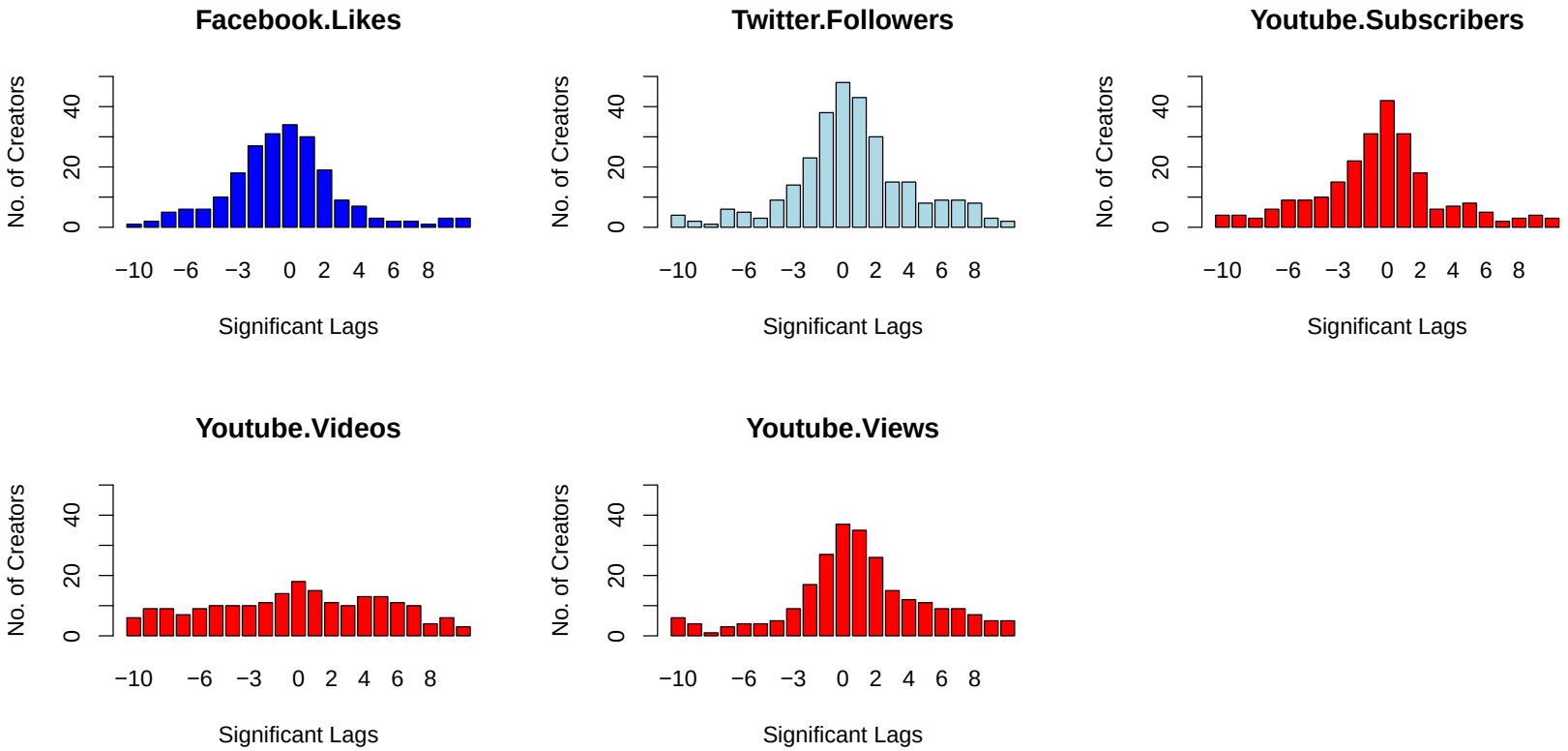


Figure: Cross-correlation between linear trend residuals of variables. The plots are histograms of significant lags for 103 creators using `ccf` function in **R**. For our forecast, we need correlation for lag −4 or less.

Model with Quadratic Trend

Model for `Patrons`, Y, and an underlying AR(1) time series:

$$y_t = \beta_0 + \beta_1 t + \beta_2 t^2 + \alpha(y_{t-1} - \beta_0 - \beta_1(t-1) - \beta_2(t-1)^2) + \epsilon_t \tag{14}$$

Assumed prior distribution for the parameters:

$$\beta_0 \sim \text{Uniform}(0, 10000) \tag{15}$$
$$\beta_1 \sim \text{Normal}(0, 100^2) \tag{16}$$
$$\beta_2 \sim \text{Normal}(0, 10^2) \tag{17}$$
$$\log \sigma \sim \text{Uniform}(-5, 5) \tag{18}$$
$$\alpha \sim \text{Normal}(0, 1^2) \tag{19}$$

The coefficient for quadratic trend was significant for less than 50% creators.

Quantiles[1] of DNest4 Results for Linear Trend

RMSE is quite low for c.50% of the creators.

Quantiles[1]	Accuracy[2]	RMSE[3]	CI^4_{t+1}	CI^4_{t+4}
0%	95.00	12.05	3.47	7.63
10%	98.41	28.13	5.02	11.82
25%	99.45	48.71	6.75	15.01
50%	99.84	99.25	9.52	19.64
75%	99.95	236.94	14.64	29.07
90%	99.98	502.15	22.31	41.84
100%	100.00	1156.53	39.63	69.49

[1] Quantiles are over 103 creators.

[2] Accuracy for 4 months' test data $= 100 * \left(1 - SSE / \sum_{h=1}^{4} y_{t+h}^2\right)$

[3] RMSE is the Root Mean Square Error.

[4] CI_{t+h} is the width of 95% credible interval as a percentage of $\hat{y}_{t+h}$

Ratio of Results - DNest4 linear trend vs. `auto.arima`

Bayesian linear trend model is better for c.50% of the creators.

Quantiles[1]	Accuracy[2]	RMSE[3]	CI^4_{t+1}	CI^4_{t+4}
0%	0.95	0.17	0.20	0.17
10%	0.99	0.62	0.51	0.36
25%	1.00	0.86	0.86	0.74
50%	1.00	1.07	1.02	1.04
75%	1.00	1.61	1.12	1.18
90%	1.00	2.58	1.28	1.41
100%	1.01	8.43	1.50	2.29

[1] Quantiles are over 103 creators.

[2] Ratio of accuracies of the two models, DNest4 : `auto.arima`

[3] Ratio of RMSE (Root Mean Square Error).

[4] Ratio of CI_{t+h} i.e. the width of 95% credible interval as a percentage of $\hat{y}_{t+h}$

Ratio of DNest4 Results - Log Growth vs. Linear Trend

Prediction intervals are wider for log growth model.

Quantiles[1]	Accuracy[2]	RMSE[3]	CI^4_{t+1}	CI^4_{t+4}
0%	0.98	0.09	0.69	1.14
10%	0.99	0.37	0.99	1.96
25%	1.00	0.64	1.25	2.71
50%	1.00	1.05	1.65	4.03
75%	1.00	1.62	2.70	7.20
90%	1.01	2.50	4.82	29.52
100%	1.05	6.25	132336.41	491744.23

[1] Quantiles are over 103 creators.

[2] Ratio of accuracies of the Bayesian DNest4 models, Log Growth : Linear Trend

[3] Ratio of RMSE (Root Mean Square Error).

[4] Ratio of CI_{t+h} i.e. the width of 95% credible interval as a percentage of $\hat{y}_{t+h}$

Ratio of DNest4 Results - Quadratic vs. Linear Trend

Model with quadratic trend performed better for more than 25% creators.

Quantiles[1]	Accuracy[2]	RMSE[3]	CI^4_{t+1}	CI^4_{t+4}
0%	0.89	0.11	0.64	0.73
10%	0.99	0.33	0.83	0.87
25%	1.00	0.75	0.92	1.03
50%	1.00	1.00	1.05	1.27
75%	1.00	1.49	1.18	1.54
90%	1.01	2.82	1.33	1.87
100%	1.05	12.25	1.81	6.64

[1] Quantiles are over 103 creators.

[2] Ratio of accuracies of the Bayesian DNest4 models, Quadratic : Linear trends

[3] Ratio of RMSE (Root Mean Square Error).

[4] Ratio of CI_{t+h} i.e. the width of 95% credible interval as a percentage of $\hat{y}_{t+h}$

Conclusions

- Bayesian DNest4 model with linear trend performed better or similar for 50% of the creators compared to `auto.arima`
  - `auto.arima` fit a unique classical time series model to each of the 103 creators.
  - In contrast, the same Bayesian model equation was used to fit all of them.
- The linear trend model also performed better than other Bayesian models.
- Quadratic trend is better suited to more than 25% of the creators.

Further Directions

- Group similar time series to fit a few models instead of one.
  - E.g. linear trend for 50% of the creators, quadratic trend for another 25% and so on.
- Hierarchical model to capture trend and fluctuations common to all creators.
- Investigate effect of new publications and external factors on spikes in variance.
- Model shifts in trend.
  - E.g. using a binomial distribution to model new publications.
  - Get data on past and future publications/releases, policy changes, etc.

Internet References I

1. Patreon `https://www.patreon.com/`
2. Graphtreon `https://graphtreon.com/patreon-stats`
3. Patreon.com profile example: Amanda Palmer
   `https://www.patreon.com/amandapalmer`
4. Graphtreon.com data example: Amanda Palmer
   `https://graphtreon.com/creator/amandapalmer`
5. Hyndman, R. **R** package forecast: Forecasting Functions for Time Series and Linear Models, 2018
   `https://cran.r-project.org/web/packages/forecast/index.html`
6. Brewer, B.J. Diffusive Nested Sampling software, 2018
   `https://github.com/eggplantbren/DNest4`

Internet References II

7. Brewer, B. J., Foreman-Mackey, D., 2018. Dnest4: Diffusive nested sampling in C++ and Python. Journal of Statistical Software Vol. 86 (7). https://www.jstatsoft.org/article/view/v086i07
8. The R Project for Statistical Computing https://www.r-project.org/
9. R Studio software https://www.rstudio.com/
10. TeXstudio software https://www.texstudio.org/
11. GitHub link to Saurabh's code used in this dissertation https://github.com/nzsaurabh

Thank You!