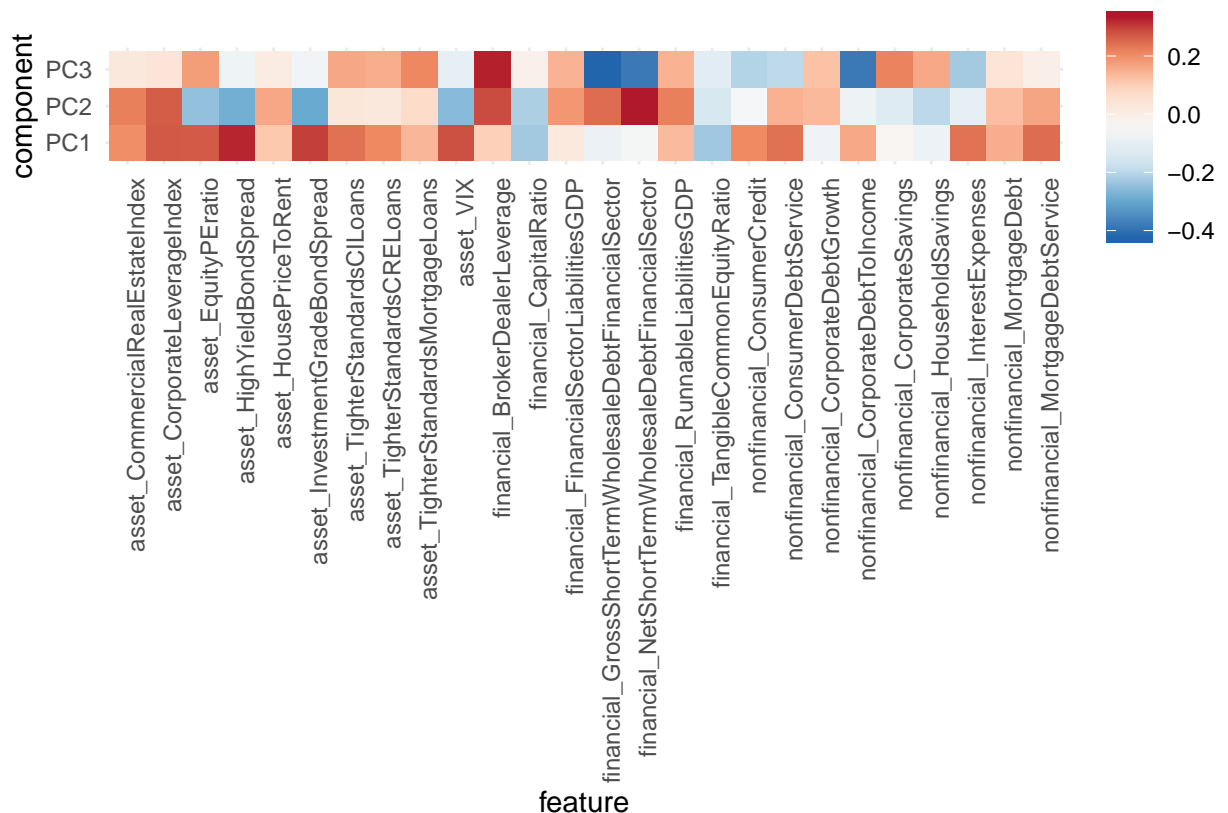


# Thesis Update and Results - 1/2020

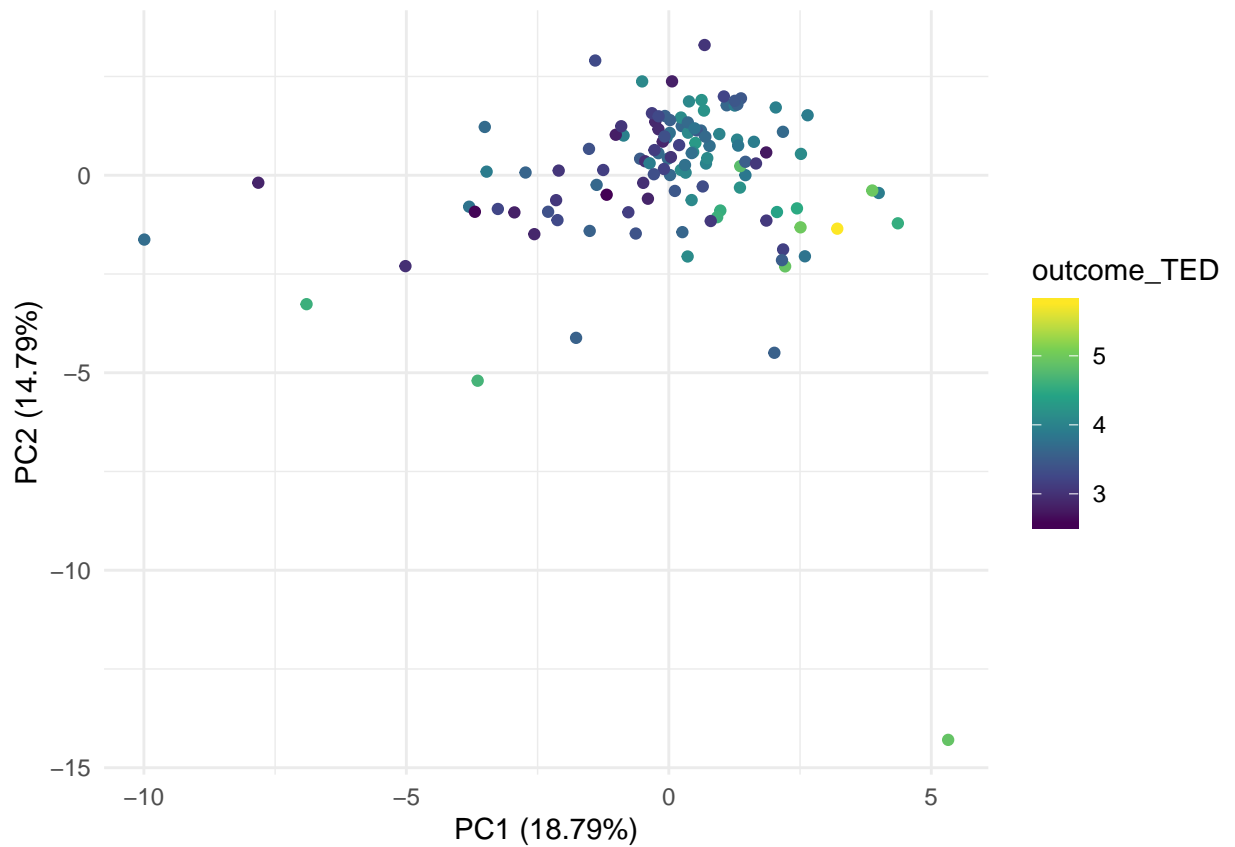
## Kiley dataset

Recall the Kiley dataset originally had 118 observations of 26 financial stability indicators measured quarterly from Q1 1990 to Q2 2019. After diff'ing to assure stationarity, we're left with 117 differenced observations representing the quarter-on-quarter change for each indicator. We first perform PCA, which should give an idea of the indicators that contribute most to the variance within the dataset.

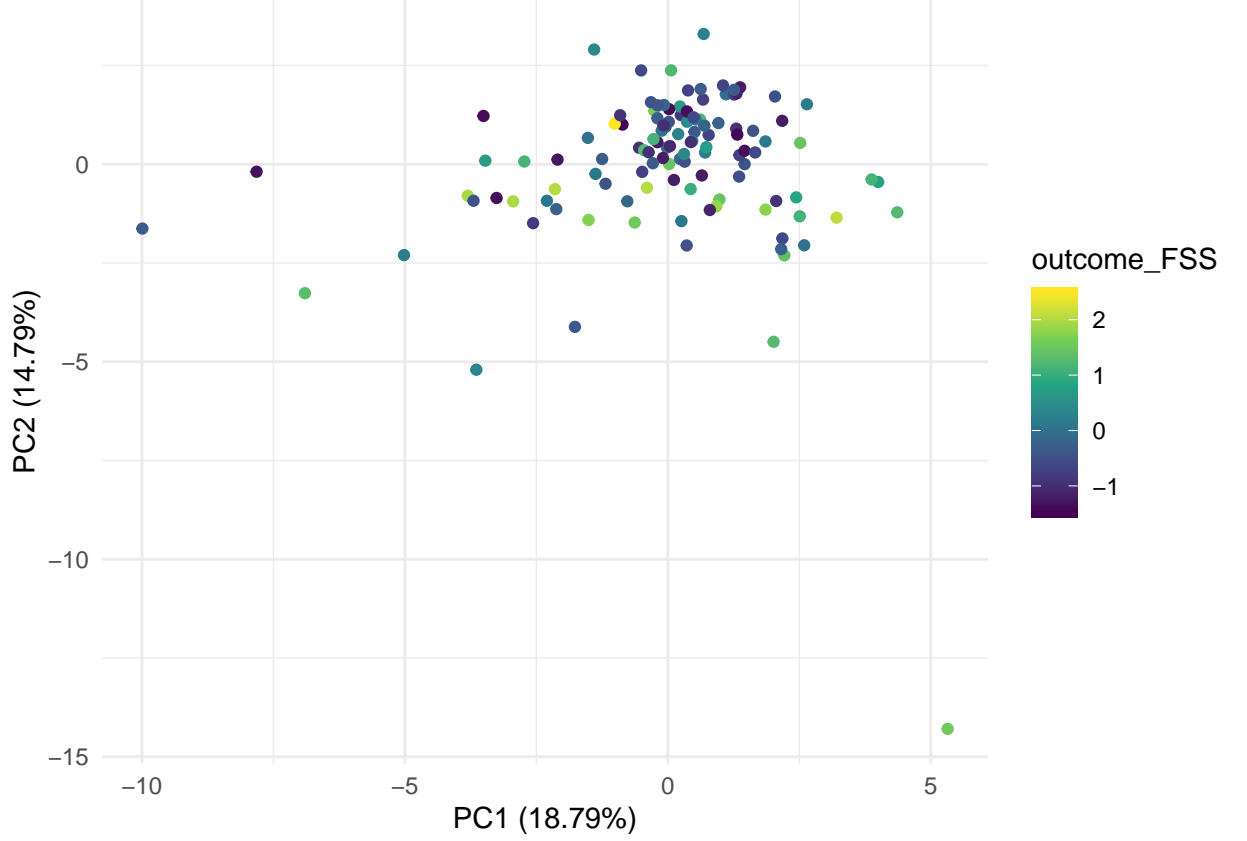


The heatmap is interesting – looks like the first PC (which explains ~20% of the variance) is highly correlated with asset pricing variables, and the second two have high loadings on the financial sector variables.

We now introduce our two outcome variables, the TED spread and the FSS index. The TED spread is a measure of bank funding liquidity that quantifies the cost of interbank lending, and we hypothesize can be predicted by strain on the financial system, the idea being that stress to the financial system in period  $t-1$  causes decreased liquidity and therefore an elevated TED spread in period  $t$ . If this is the case – i.e., if the information in the Kiley dataset is accurately explained by the first two PCs, and the variance explained by the first two PCs corresponds to variance in the TED spread, we expect to see a clear pattern in the scatterplot here. This would in turn hint that Kiley's indicators are predictive of the TED spread.



This is promising, as it looks like the log TED spread tends to increase with the first PC. We now introduce the FSS index, a measure of the quarterly “financial stability sentiment” of the FOMC generated by applying natural language processing techniques to the March, June/July, September, and December FOMC minutes using a dictionary and scoring system described by Londono et al. Per Londono et al, positive FSS scores indicates deteriorating Fed sentiment as to the health of the financial system. We now color the scatterplot of the first two PCs in Kiley by the scaled FSS score in the next period.



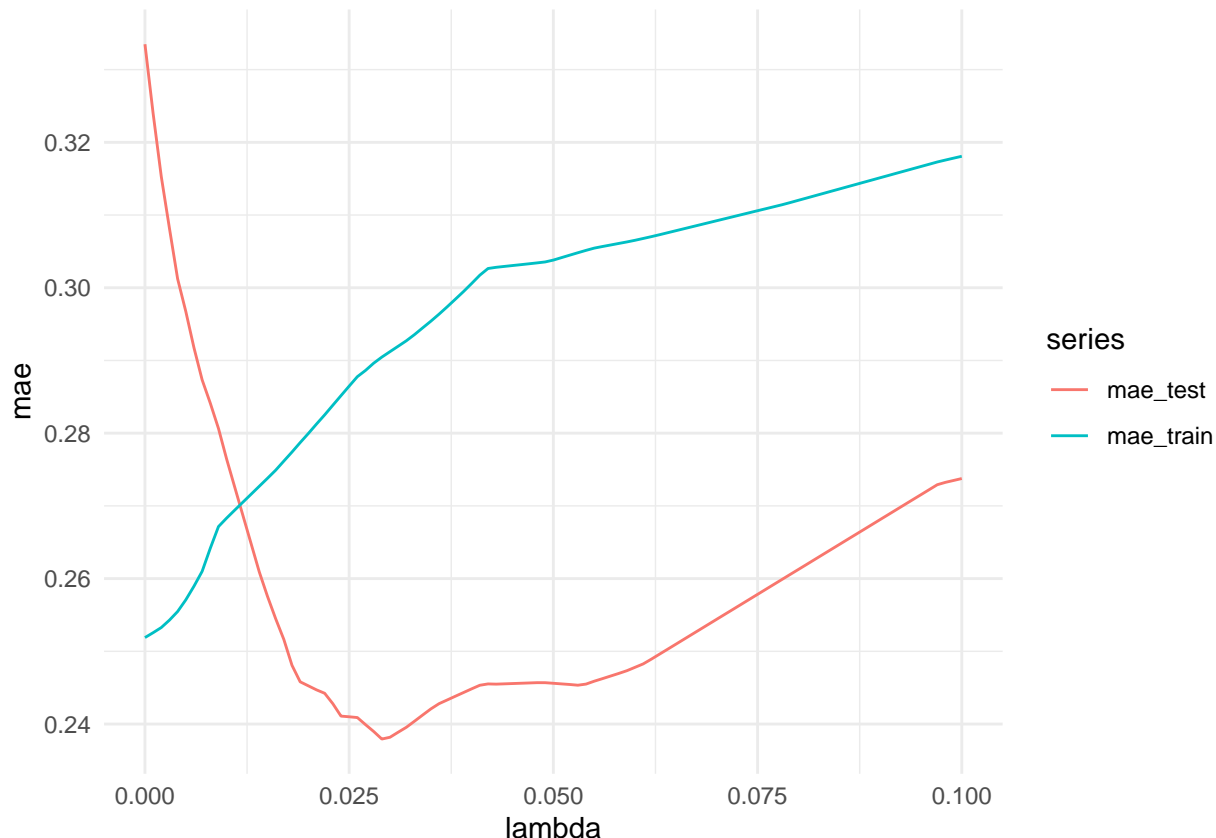
This plot is less clear, though we might see some variation in the FSS score with PC2. Either way, it looks relatively promising to continue estimating the models.

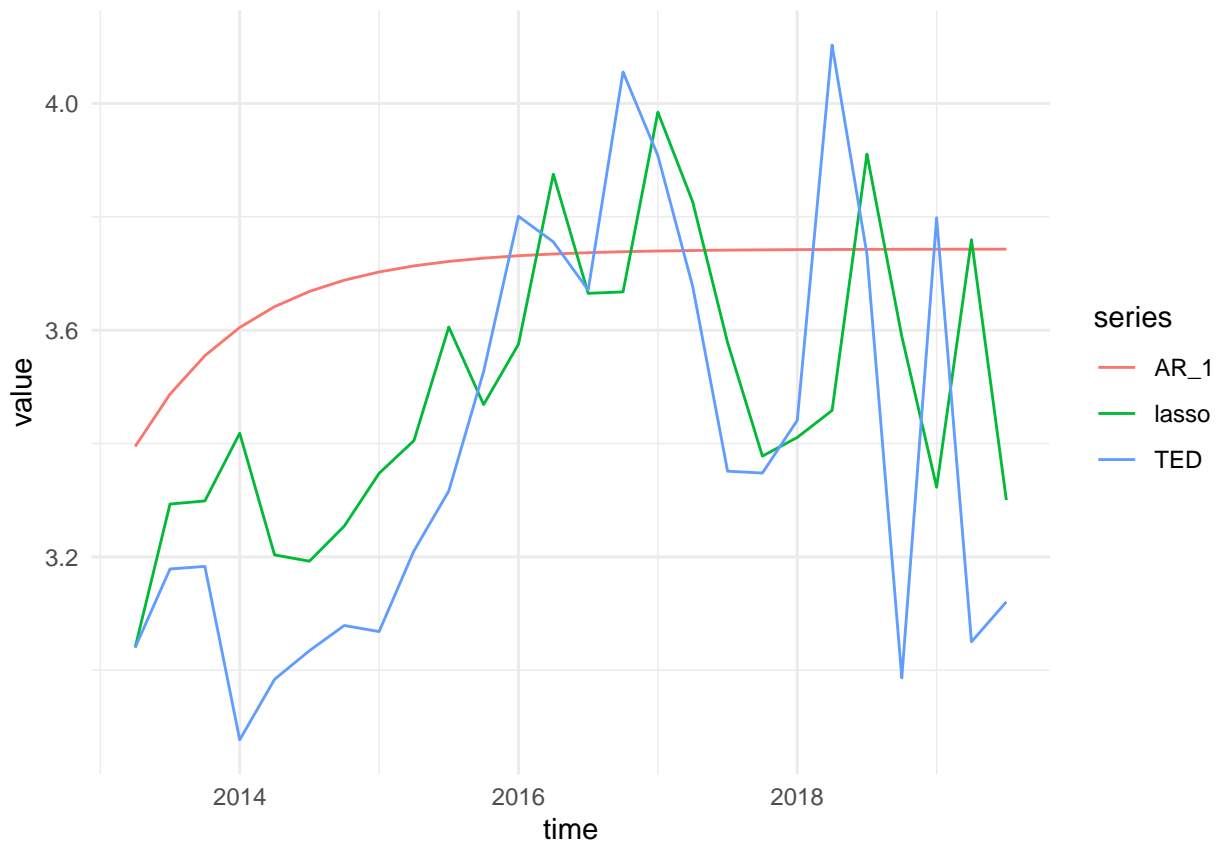
### Kiley - TED spread

Now we'd like to quantify the effect of each of Kiley's indicators on liquidity and the Fed's financial sentiment. We have two goals: first, we obviously want to accurately predict liquidity in the period ahead given the state of the financial system in the current period; second, we want this predictive model to be interpretable, i.e. we want the model to identify which of the 26 features in Kiley's dataset carry the most predictive power. The traditional regression framework cannot be accurately applied here as our dataset is high-dimensional and highly correlated, and the TED spread is itself a (stationary) time series with an autoregressive component. Therefore, we estimate an autoregressive LASSO model, regularizing the coefficients in order to perform inline feature selection in addition to getting the best predictions. We start with the model for the TED spread, specified as follows:

$$\log(y_t) = \mu + \phi_1 \log(y_{t-1}) + \Lambda \mathbf{X}_{t-1} + \epsilon_t$$

where we're modeling the log of the TED spread  $y_t$  in period  $t$  given the log TED spread in period  $t - 1$  and a vector of differenced indicators  $\mathbf{X}_{t-1}$  reflecting the change in each series in Kiley's dataset from period  $t - 2$  to period  $t - 1$ . We choose a training set of observations from Q3 1990 to Q4 2012 (the first 90 observations we can use, after differencing) and test on Q1 2013 to Q2 2019 (the last 26) – this corresponds to about an 80/20% train/test split. To get an idea of the LASSO model's performance, we first fit an ARIMA baseline with the `auto.arima` function to ensure we get the best time series model for comparison, and then we score the models on the mean absolute error of the log TED spread on the test set.





```
## Warning: Expected 2 pieces. Additional pieces discarded in 1 rows [1].
```

```
# LASSO model R2
```

```
(r2 = lasso1$dev.ratio[which.min(mae_test)])
```

```
## [1] 0.6422264
```

```
# pct improvement over AR(1), log bp
```

```
(pct_improvement_log = 100* (mae_ar - min(mae_test)) / mae_ar)
```

```
## [1] 36.17519
```

The LASSO model provides good fit at a reasonable  $\lambda$  value (shown in the train/test error plot over 100 values), selecting 13 nonzero features and the lagged log TED spread. The predictions appear reasonable and we see clear improvement in predictive performance over the AR(1) baseline. Good start. Will leave the interpretation of the coefficients for a separate section.

Table 1:

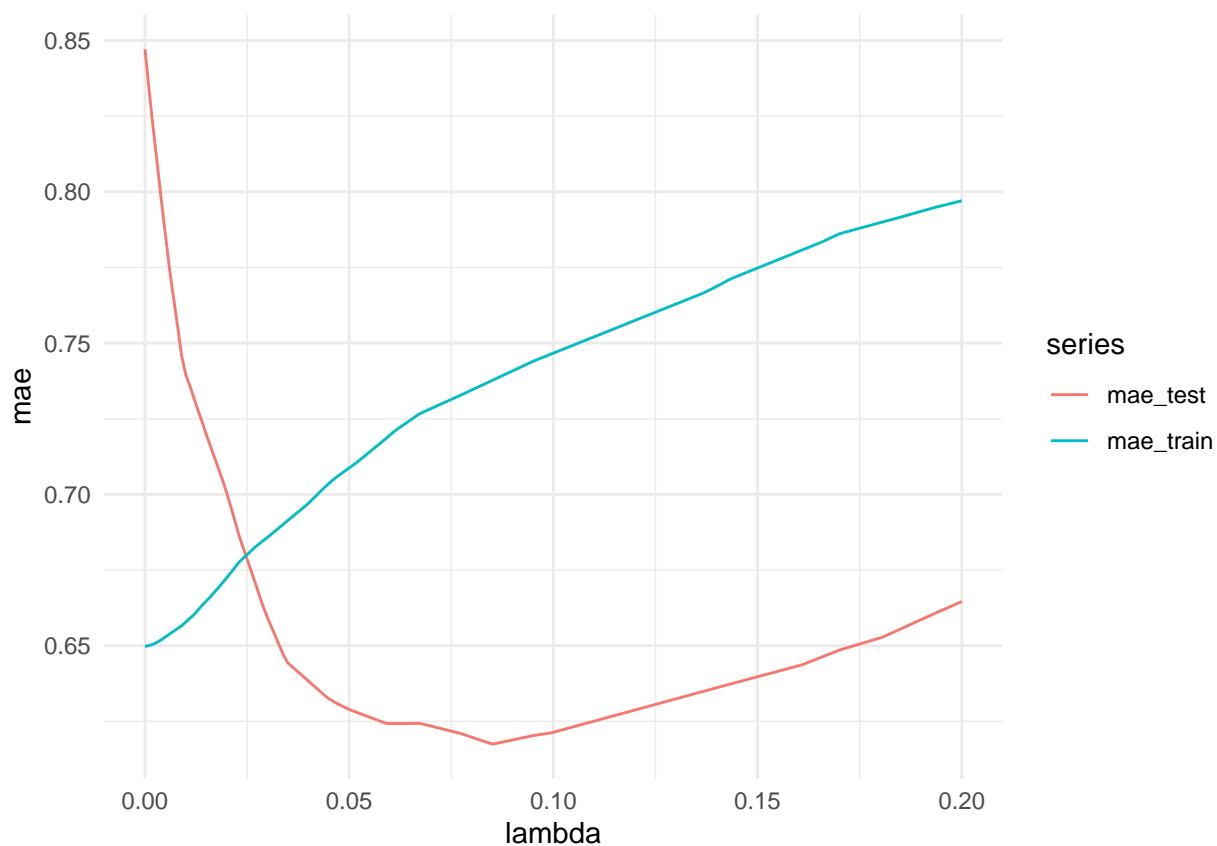
Class	Indicator	Coefficient
	Intercept	1.631
	lagTED	0.559
financial	TangibleCommonEquityRatio	-0.096
asset	TighterStandardsCRELoans	0.076
asset	HousePriceToRent	-0.072
nonfinancial	ConsumerCredit	-0.039
nonfinancial	ConsumerDebtService	0.034
asset	CorporateLeverageIndex	0.024
financial	BrokerDealerLeverage	-0.022
nonfinancial	CorporateDebtGrowth	0.021
nonfinancial	MortgageDebt	-0.017
nonfinancial	HouseholdSavings	-0.013
asset	HighYieldBondSpread	0.012
financial	FinancialSectorLiabilitiesGDP	-0.011
asset	TighterStandardsMortgageLoans	0.002
financial	GrossShortTermWholesaleDebtFinancialSector	0
financial	NetShortTermWholesaleDebtFinancialSector	0
financial	RunnableLiabilitiesGDP	0
financial	CapitalRatio	0
nonfinancial	CorporateSavings	0
nonfinancial	CorporateDebtToIncome	0
nonfinancial	InterestExpenses	0
nonfinancial	MortgageDebtService	0
asset	InvestmentGradeBondSpread	0
asset	EquityPERatio	0
asset	CommercialRealEstateIndex	0
asset	TighterStandardsCILoans	0
asset	VIX	0

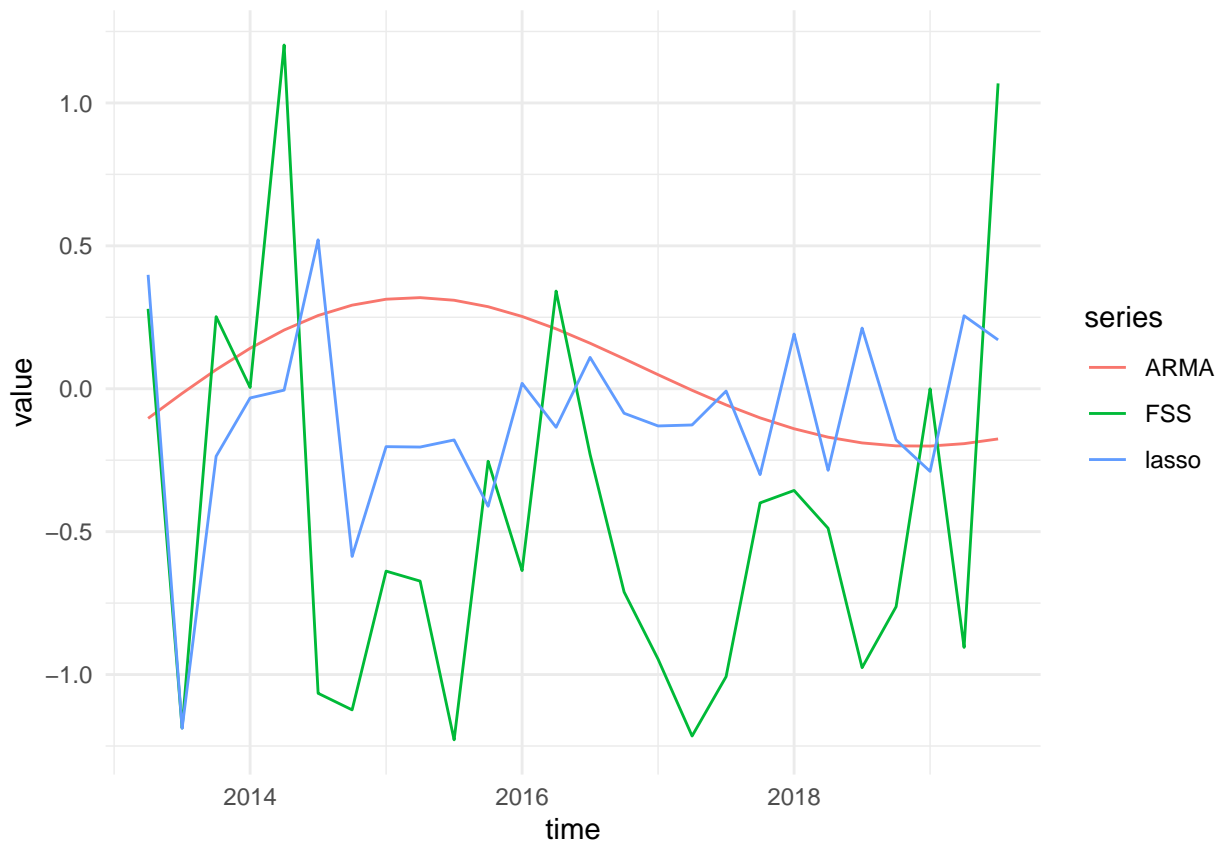
## Kiley - FSS index

Next, we want to see if the same indicators are predictive of the Fed's financial stability sentiment – if this is the case, we can have confidence that the lasso model selected valuable features that inform financial system strain as measured in two completely separate ways. We therefore specify the FSS model as follows:

$$s_t = \nu + \gamma_1 s_{t-1} + \Theta \mathbf{X}_{t-1} + \epsilon_t$$

where, similarly, we're modeling the (scaled to zero mean/unit variance) FSS score  $s_t$  in period  $t$  given the scaled score in period  $t - 1$  and the same vector of differenced indicators  $\mathbf{X}_{t-1}$  reflecting the change in each series in Kiley's dataset from period  $t - 2$  to period  $t - 1$ . We fit the LASSO by the same procedure and produce the same model diagnostics:





```
stargazer(df_coef, summary = FALSE, rownames = FALSE, header = FALSE)
```

```
# FSS model R2
```

```
(r2 = lasso1$dev.ratio[which.min(mae_test)])
```

```
## [1] 0.2659562
```

```
# FSS model pct improvement over ARMA(1,1)
```

```
(pct_improvement = 100* (mae_ar - min(mae_test)) / mae_ar)
```

```
## [1] 17.04949
```

This is where things start breaking down. Our model fit clearly decreases compared to the TED spread, which we can notice immediately in two ways: first, the  $R^2$  is only 0.27, which while not at the level of uninterpretable certainly doesn't give much confidence in the model results. I think we have a very good visualization of the bias-variance tradeoff here, in kind of an unfortunate way, because as we increase  $\lambda$  in order to prevent overfitting and reduce the variance of the estimator, the model's bias increases such that we lose overall fit. Basically, if we chose a smaller  $\lambda$  as in the last model, we'd have a higher  $R^2$ , so we could consider optimizing the regularization based on that instead of test error (although the  $R^2$  increases



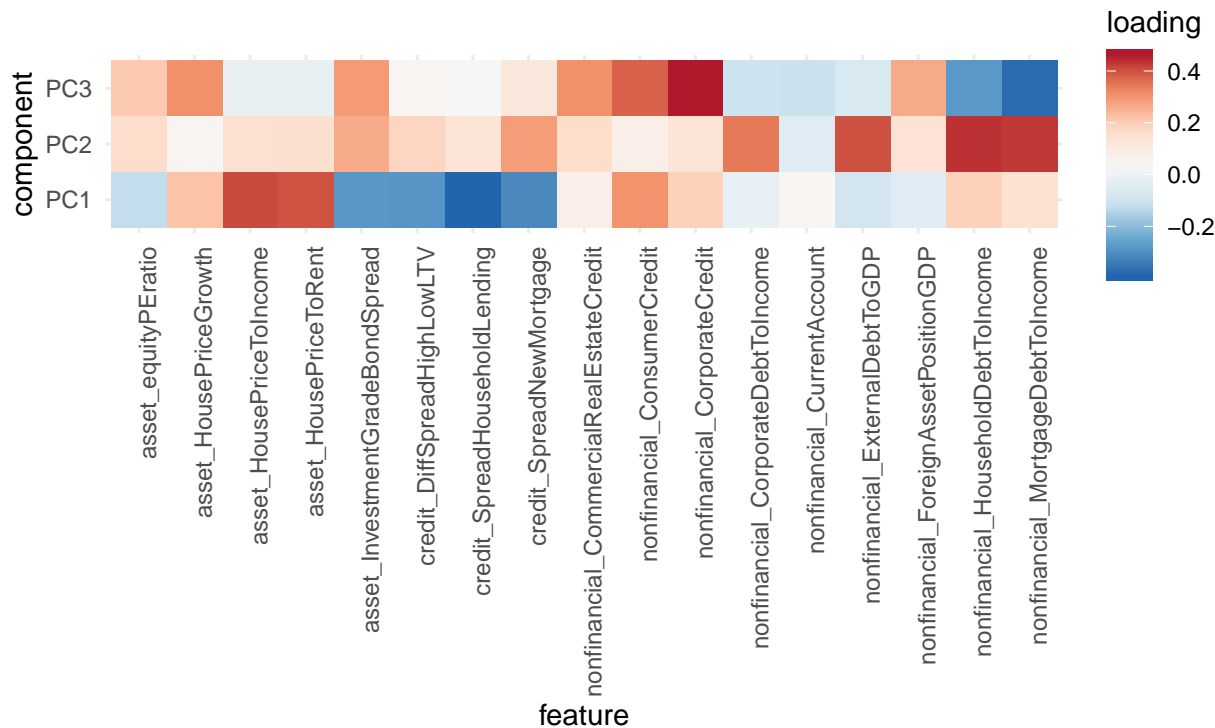
Table 2:

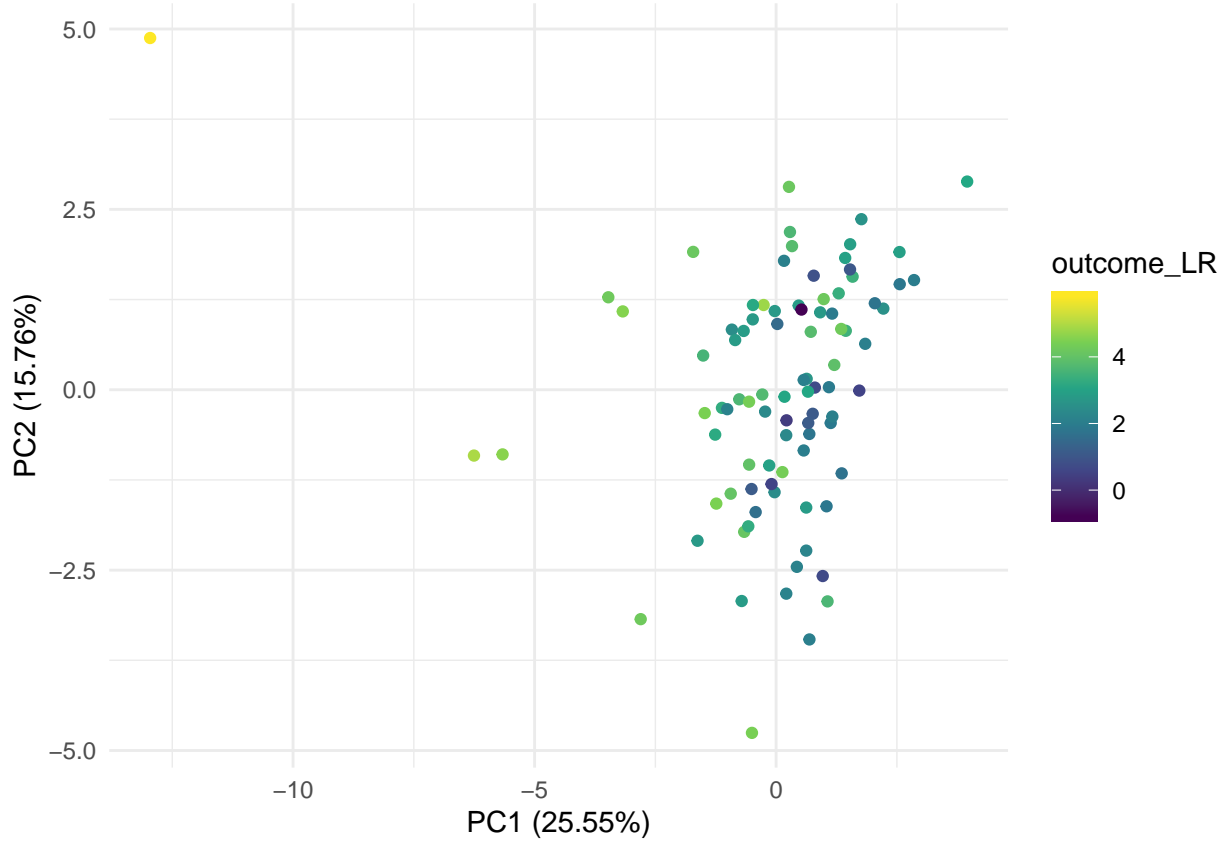
Class	Indicator	Coefficient
	lagFSS	0.232
nonfinancial	HouseholdSavings	0.18
asset	CorporateLeverageIndex	-0.126
asset	TighterStandardsMortgageLoans	0.117
nonfinancial	MortgageDebt	0.075
	Intercept	0.054
financial	TangibleCommonEquityRatio	-0.013
financial	GrossShortTermWholesaleDebtFinancialSector	0
financial	BrokerDealerLeverage	0
financial	NetShortTermWholesaleDebtFinancialSector	0
financial	RunnableLiabilitiesGDP	0
financial	CapitalRatio	0
financial	FinancialSectorLiabilitiesGDP	0
nonfinancial	CorporateSavings	0
nonfinancial	ConsumerCredit	0
nonfinancial	ConsumerDebtService	0
nonfinancial	CorporateDebtToIncome	0
nonfinancial	CorporateDebtGrowth	0
nonfinancial	InterestExpenses	0
nonfinancial	MortgageDebtService	0
asset	InvestmentGradeBondSpread	0
asset	EquityPERatio	0
asset	HousePriceToRent	0
asset	HighYieldBondSpread	0
asset	CommercialRealEstateIndex	0
asset	TighterStandardsCILoans	0
asset	VIX	0
asset	TighterStandardsCRELoans	0

monotonically as we add more variables to the model, meaning it strictly decreases as we increase  $\lambda$ , so it might not actually be a meaningful parameter to score the model on). Note that we can't compare the test MAE in the TED spread model to this one, as the units are different, but the percent improvement should provide a consistent basis for comparison. Accordingly, we only get 17% improvement over the ARIMA baseline here, and the selected features don't seem to correlate well to the ones selected for the TED spread model. Again leaving interpretation to later, but these results don't seem ideal.

### Aikman - LIBOR-Repo spread

As a robustness check, we now introduce an analog of the first model from a different economy. Aikman et al. have collected a dataset of financial stability indicators for the UK, which though more sparse than Kiley's in terms of both features and observations, still contains complete data for 90 quarterly observations of 16 features from Q1 1996 to Q2 2019. For our outcome variable, we use the UK analog of the TED spread, the spread between the 3-month LIBOR and Repo rate on 3-month UK gilts. Since we have a new dataset of indicators, we first difference for stationarity and perform PCA, similarly to our preliminary analysis on Kiley's dataset:





Looks OK so far – we see one clear outlier in the scatterplot, but see slightly more variance explained by the first two PCs, a good sign. The heatmap looks reasonable (credit and certain asset pricing variables have the most influence on PC1, and nonfinancial sector variables on the other two). Let's now fit the following model, essentially a repeat of the Kiley-TED spread specification:

$$\log(r_t) = \alpha + \theta_1 \log(r_{t-1}) + \Omega \mathbf{A}_{t-1} + \epsilon_t$$

where we're modeling the log of the LIBOR-Repo spread  $r_t$  in period  $t$  given the log LR spread in period  $t - 1$  and a vector of differenced indicators  $\mathbf{X}_{t-1}$  reflecting the change in each series in Aikman's dataset from period  $t - 2$  to period  $t - 1$ .

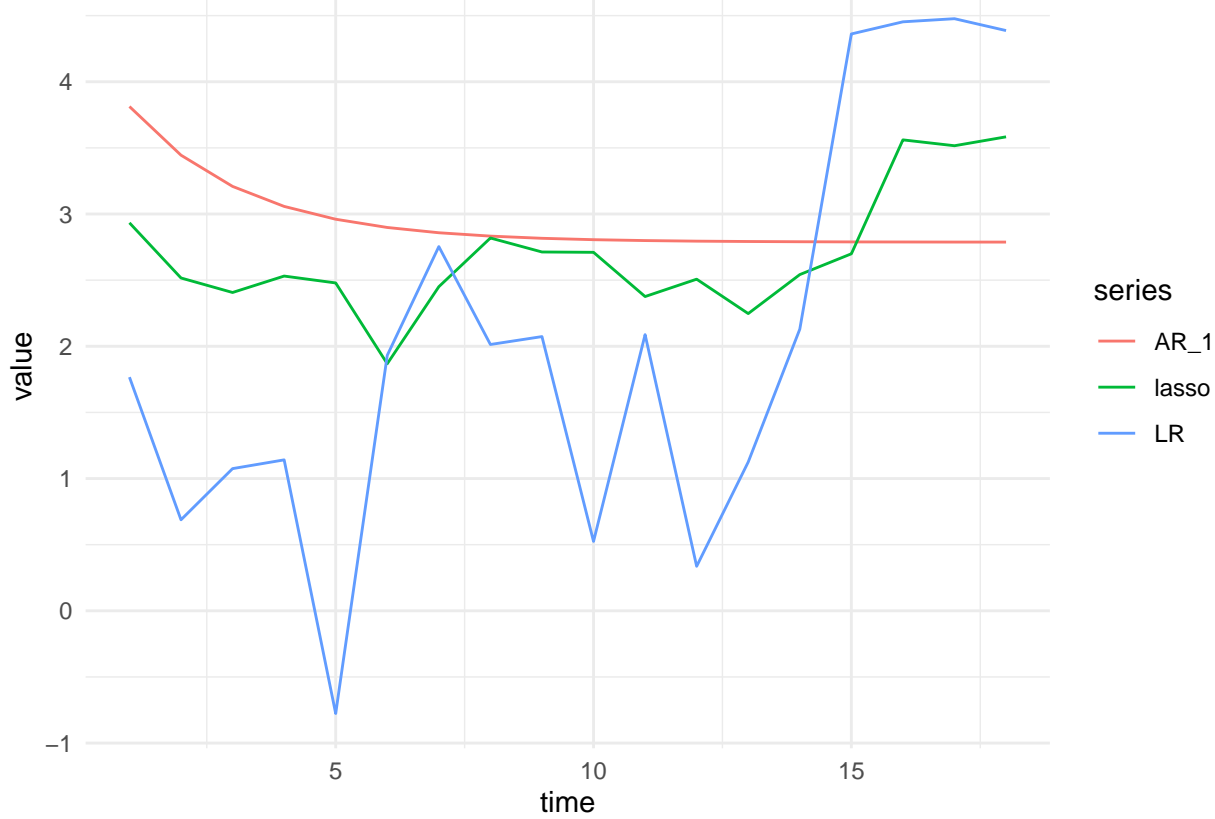
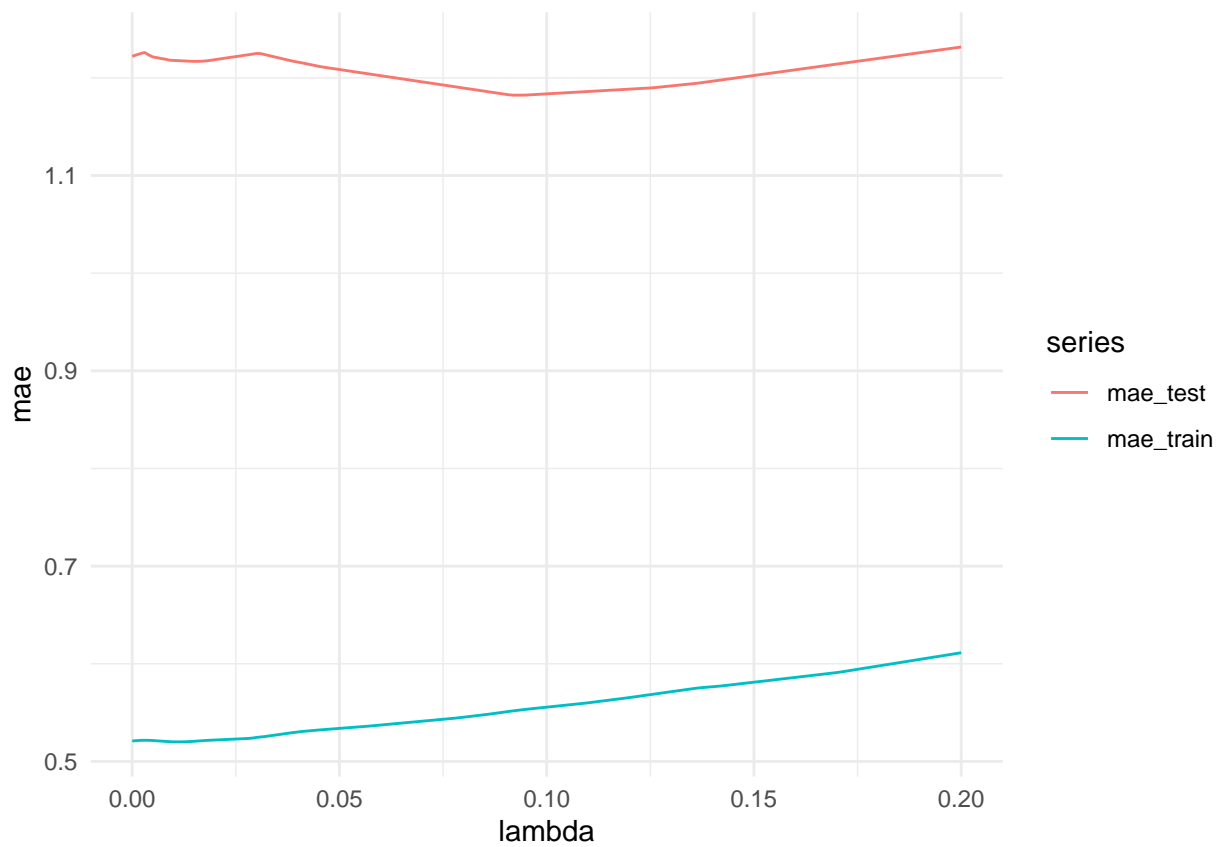


Table 3:

Class	Indicator	Coefficient
	Intercept	2.192
	lagLR	0.275
asset	InvestmentGradeBondSpread	0.183
nonfinancial	ConsumerCredit	-0.127
nonfinancial	CorporateCredit	-0.109
asset	HousePriceToRent	-0.088
nonfinancial	CommercialRealEstateCredit	0.049
asset	equityPERatio	0.002
nonfinancial	MortgageDebtToIncome	0
nonfinancial	HouseholdDebtToIncome	0
nonfinancial	CorporateDebtToIncome	0
nonfinancial	CurrentAccount	0
nonfinancial	ForeignAssetPositionGDP	0
nonfinancial	ExternalDebtToGDP	0
asset	HousePriceToIncome	0
asset	HousePriceGrowth	0
credit	SpreadNewMortgage	0
credit	DiffSpreadHighLowLTV	0
credit	SpreadHouseholdLending	0

```
(r2 = lasso1$dev.ratio[which.min(mae_test)])
```

```
## [1] 0.4868978
```

```
(pct_improvement_log = (mae_ar - min(mae_test)) / mae_ar)
```

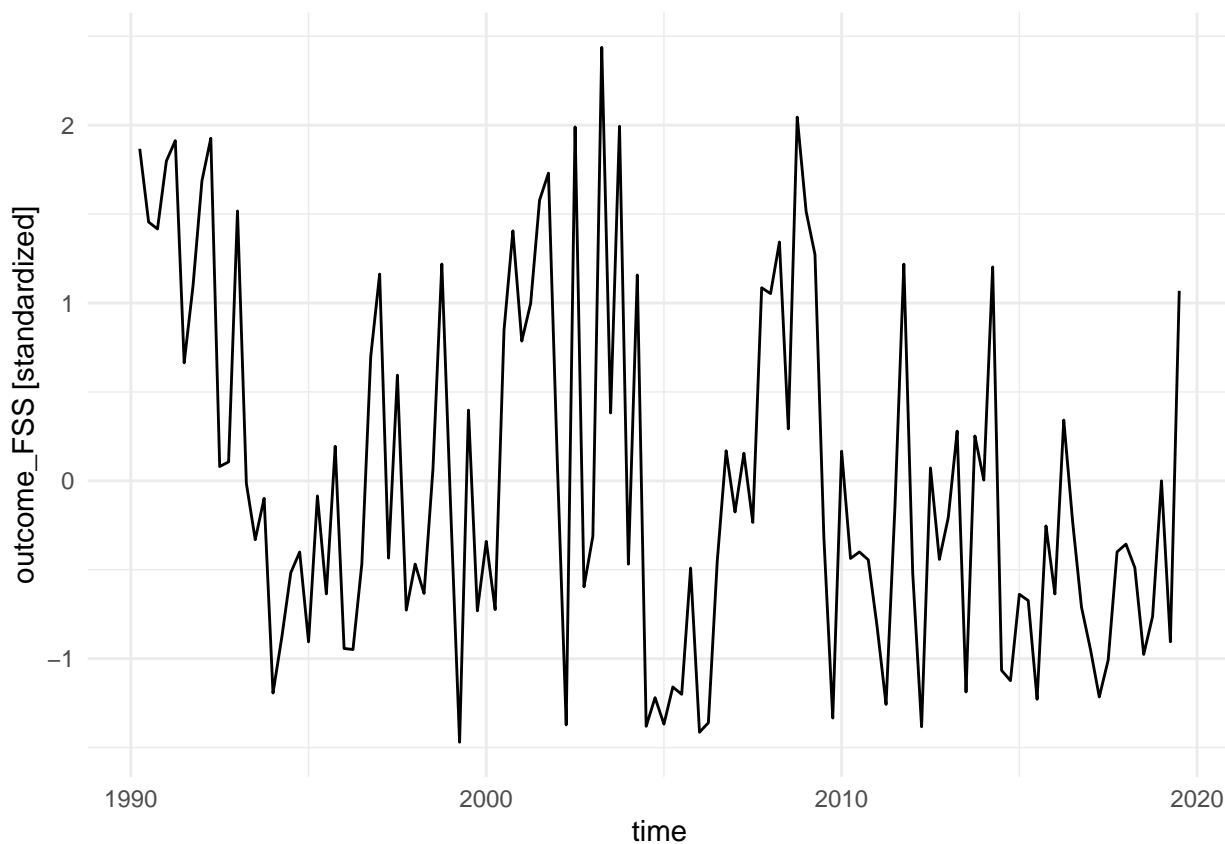
```
## [1] 0.2792723
```

The model doesn't seem to converge on this smaller dataset, even when we compensate by regularizing more severely (increasing  $\lambda$  beyond 0.1). TBD how to fix this. The other diagnostics look OK though, but I think the  $R^2$  value is an artifact of overfitting (look how much lower the training error is than the test error), and the reasonably good percent improvement might just show that the LR spread is more of a random-walk type thing that can't be modeled accurately by time series forecasting.

## Next Steps

I personally think the above is a reasonably good start, but more needs to be done to turn this into an actual good thesis. Some things to explore next might be more advanced time series modeling on the outcome variables themselves (if we buy that there is conditional heteroskedasticity in the outcome variables, which it looks like there might be for the FSS index, GARCH or SV models might be interesting to look at).

My thinking here is that generating the FSS scores was itself nontrivial, and we might want to feature that work in a slightly more significant way rather than just using it as another data source as if I just downloaded it from Bloomberg or something. Something else I'd like to try, but with no guarantees at all that it would work, are deep learning methods like LSTM neural networks. I don't think we have enough data to do this successfully, but would be interesting to try it regardless. Beyond that, I think coming up with a better validation/identification strategy that gives a little more information than just a percentage improvement over a baseline time series model on the test set, is key to understanding what's actually going on in these datasets. Another thing to consider is the noise within the FSS index, shown here:



We really need some way to denoise this signal so we get a nice smooth line, maybe with the peaks we see near the dot-com boom and the Recession. This might just not be what the data is showing us, which would be disappointing, but not sure what else we can do.