# How I made 37% annual return for 3 years using data science, machine learning, credit risk & TALF loans

Luis Miguel Sánchez Follow
Apr 19, 2019 · 24 min read ★

In this article, I will continue with my series of practical applications in the use of data science, this time, a practical application involving **data science, investment management, insurance and financial quantitative analysis** to analyze risk/reward opportunities, this time in fixed income instruments.

This exercise refers to the opportunity generated by the Term Asset Loan Facility (or "TALF") back in early 2009. The TALF was a government program designed to lend up to USD $1 trillion to qualified investors at very low interest rates and with the ultimate intention of jump starting the economy after the financial crisis of 2008.

The data for the exercise explained herein are derived from the prevalent market conditions in early February 2009, right after the TALF was created. The first loan disbursements from the Federal Reserve and US Treasury to qualified investors were available as of late February.

To understand the opportunity, some basic knowledge of the securitization market (Asset Backed Securities)and the origin of the financial crisis is necessary, therefore this will be explained below. If you want to learn more, a great deal of the background of the key technical elements of the financial crisis is discussed in my previous article **"Alpha Generation using Data Science & Quantitative Analysis — ABS / TALF"** which you should definitely read (here), to gain more knowledge about key elements of the process. In fact, the root of the opportunity was one this factors: the size of the ABS market before the crisis, and the world economies' link to ABS.

## How large was the securitization market leading up to the crisis? Who were the leaders in the ABS market?

The US securitization market experienced an amazing growth rate of almost 19% on an annual basis from its inception in 1985 until 2007, reaching a peak of about USD $1.2 trillion in issuance. Figure 1 shows the growth of the securitization market and its relationship to interest rates.

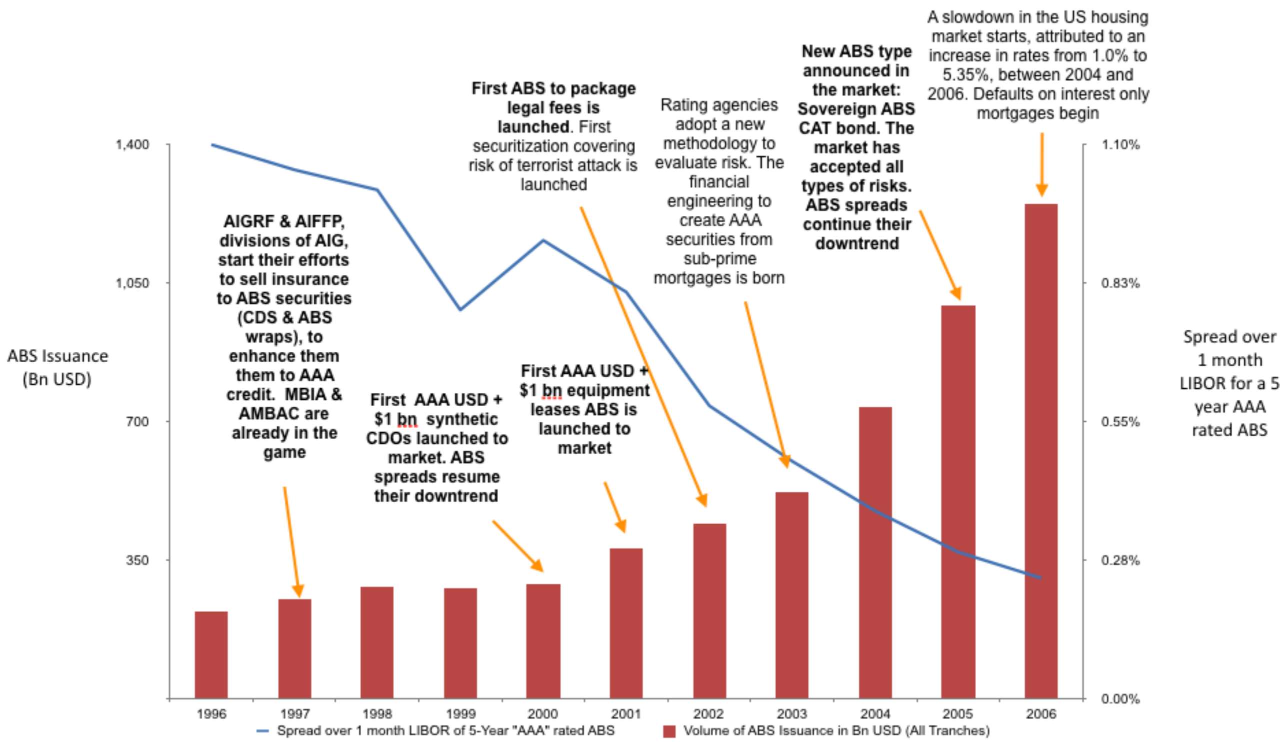
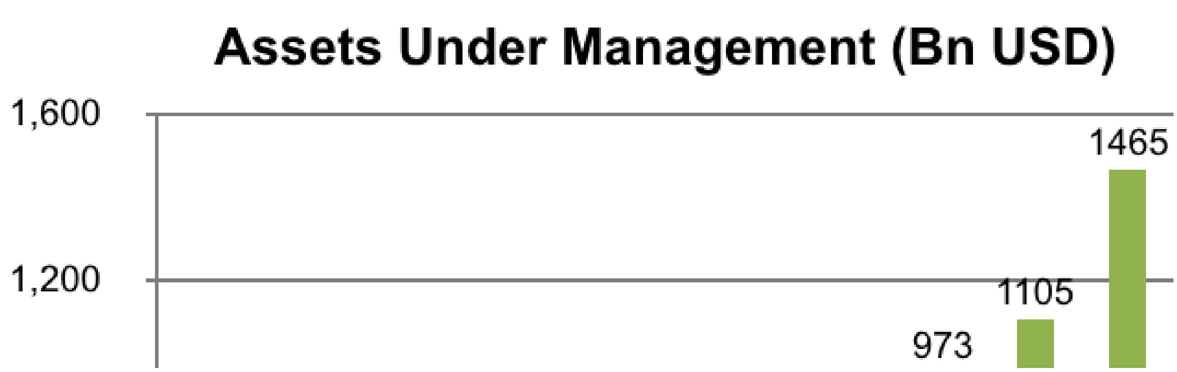



**Figure 1:** The availability of a growing supply of fixed income instruments backed by uncorrelated asset classes helped bring down financing costs.

The securitization technology developed by US quants at investment banks allowed producers to obtain cheap financing for operations and expansion. In fact, from 1985 to 2007, the average spread of a 5-year AAA rated ABS dropped from 120bps (1.20%) over LIBOR to about 20bps (0.20%) over LIBOR.



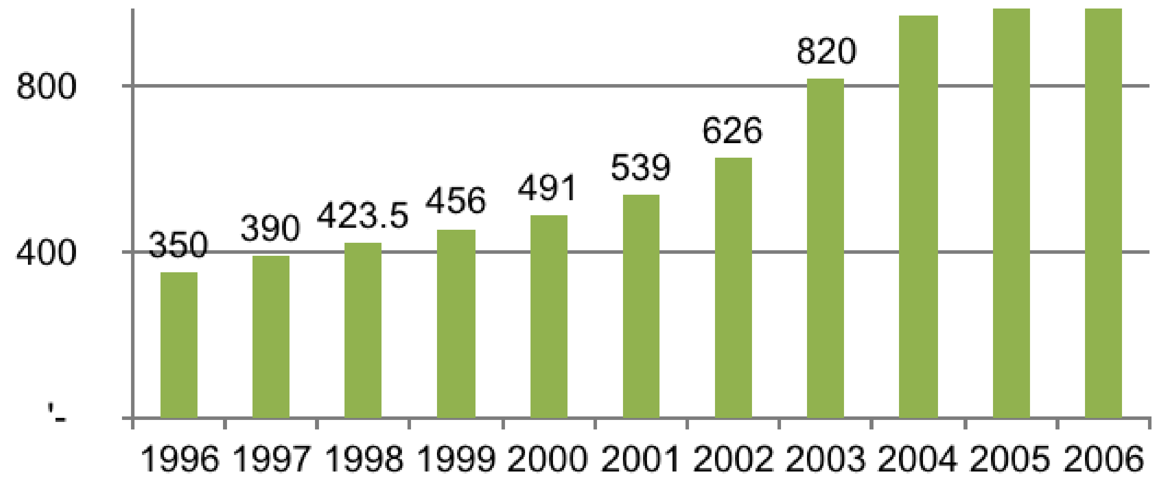



**Figure 2**: Fueling the purchase of risky pieces of Asset Backed Securities (and other securities) was the staggering amount of assets under management by hedge funds, which grew from USD $350 bn in 1996 to USD $1.4 tr in 2006 (almost 3% of the world's GDP that year). Source: SAGA Capital, LLC.

This explosion of cheap financing was supported by an incredible expansion in assets under management (AUM) of hedge funds, from about USD $350 billion in 1997 to its peak of over USD $1.4 trillion in 2006–2007 (Figure 2). Hedge funds contributed to the massive infusion of cheap capital to the world's markets, since a non- trivial number of these funds supported ABS issuance by acquiring the riskier tranches, generally the equity pieces, rated below investment grade.

At the same time AUM for hedge funds peaked, securitization reached peak volumes in 2006, with over USD $1 trillion a year in issuance. Citi, Lehman Brothers, Bank of America, JP Morgan and a few other American banks were responsible for over 1/3 of all the world's flow (Figure 3).

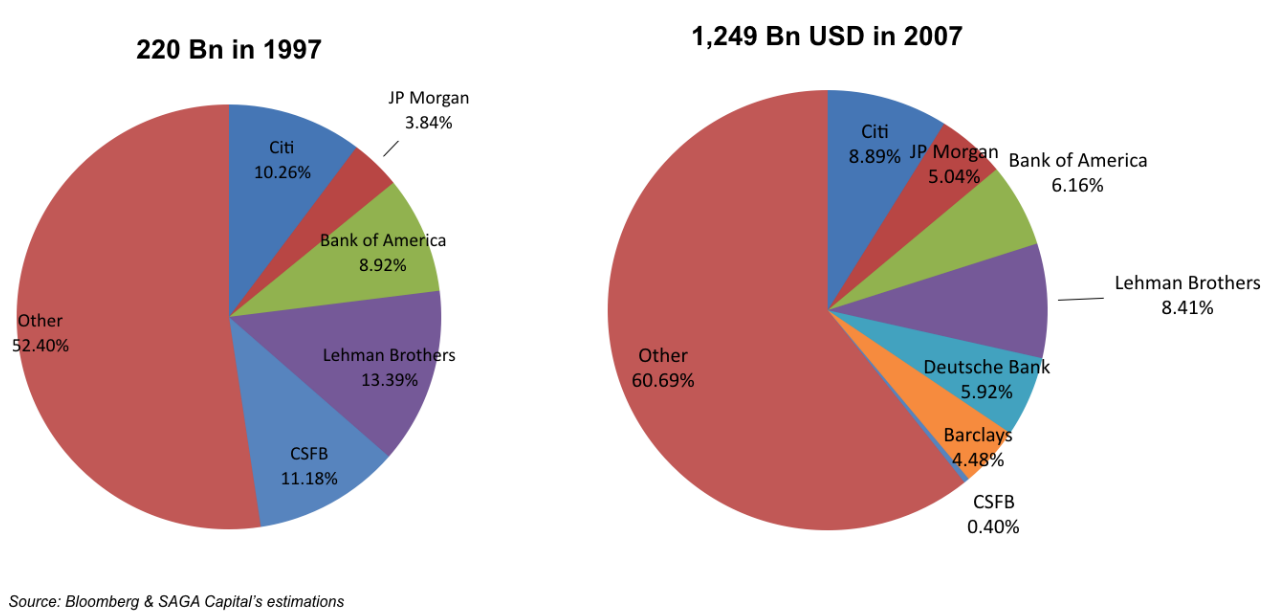


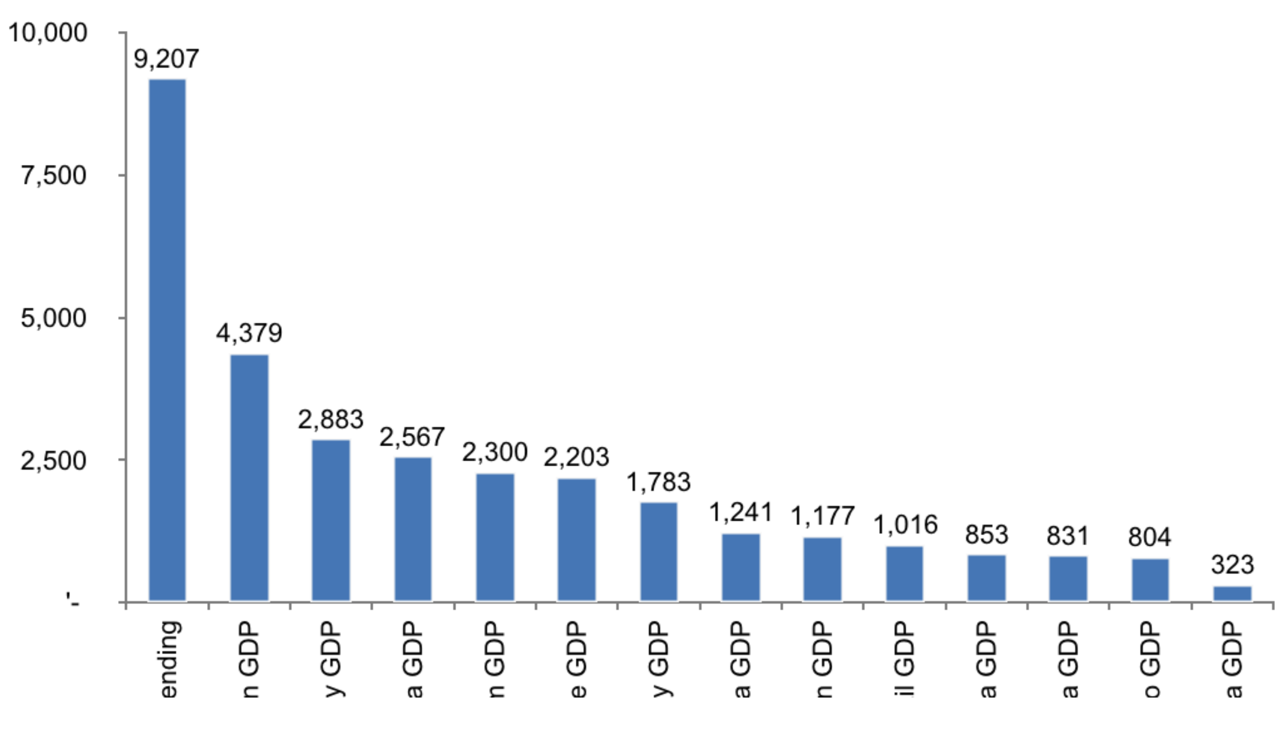
**Figure 3:** The numbers of dominant banks with the know-how and resources to create ABS grew from less than a dozen in 1996, to more than 50 in 2006. Still the largest players remained the ones shown in the chart, with Lehman Brothers among the leaders. Source: SAGA Capital, LLC.

## What was the impact of ABS on the domestic and global economies?

From a top-down approach, the securitization market contributed to the generation of new assets for monetization, and indirectly helped increase employment rates in the US and the rest of the world, as well as the expansion of domestic and global economies. At the same time the US securitization market was experiencing growth and innovation, the average US citizen dramatically increased his/her debt load, from 50.4% of Personal Disposable Income (PDI) in 1958 to over 100% of PDI during the George W. Bush administration.

In that same period, US citizens reduced their personal savings rate in favor of higher consumer spending, from a high of 11.20% of PDI during the Ronald Reagan administration, to a low of 0.25% during the George W. Bush administration.

The importance of US consumer spending in the context of the world economy and its relationship to US GDP is not well understood. The US economy in 2008 was by far the largest economy in the world, and US consumer spending represented a staggering 70% of the US GDP. To have an idea of how important it was, the chart in Figure 4 shows US Consumer Spending vs GDP of several countries.



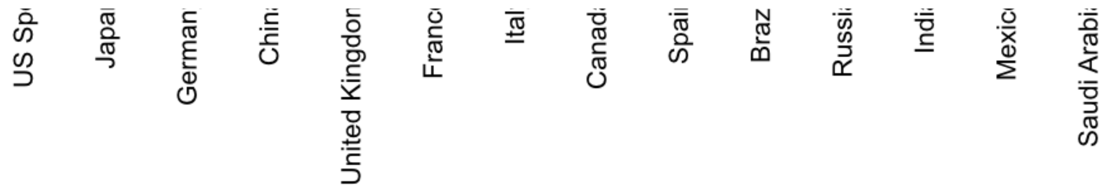



**Figure 4**: Top GPD's (2007), in billions of USD vs just one of the components of US GDP: consumer spending. Source: SAGA Capital, LLC.

In the 15 years prior to the crisis, the easy credit markets experienced in the US were fueling most of the GDP growth experienced in the US and the rest of the world, in a sort of cycle that I interpret as the cycle shown in figure 5. Industrial growth fueled by US consumer spending had a direct impact on corporate profits of world markets, as evidenced by the growth of market capitalization of virtually every stock exchange in existence.


**Figure 5:** US Consumer Spending & Securitization fueling the world's economic growth. Source: SAGA Capital, LLC.

> Easy, readily available credit to US consumers via credit cards or access to instantaneous, internet approved home equity loans encouraged consumers to save less, consume more and improve their quality of life by purchasing things like new homes, cars, a better education, and other goods and services.

In mid 2007, one type of asset class that had been growing dramatically within the securitization market started to show signs of problems. The asset class was the sub-prime mortgage, which had become very popular among some investment banks, pension funds and hedge funds, due to the supposed low risk and attractive returns of some of its tranches in a securitization.

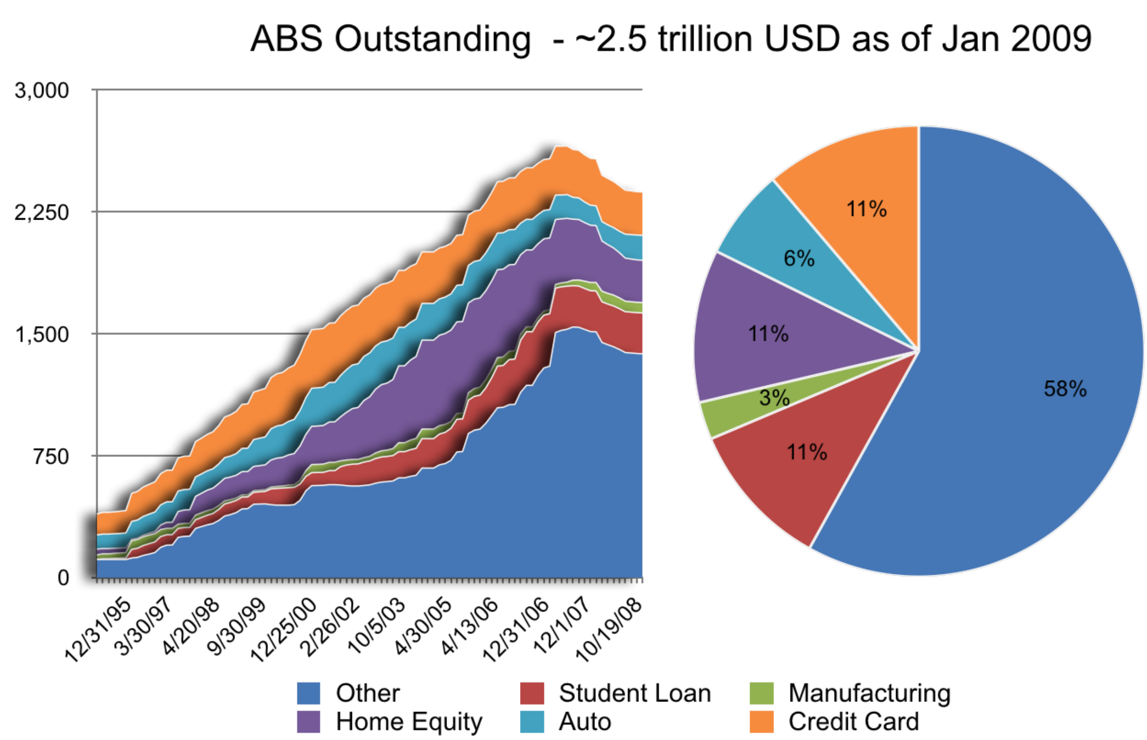



**Figure 6:** ABS Outstanding as of 2009. Source: Federal Reserve Bank of Chicago

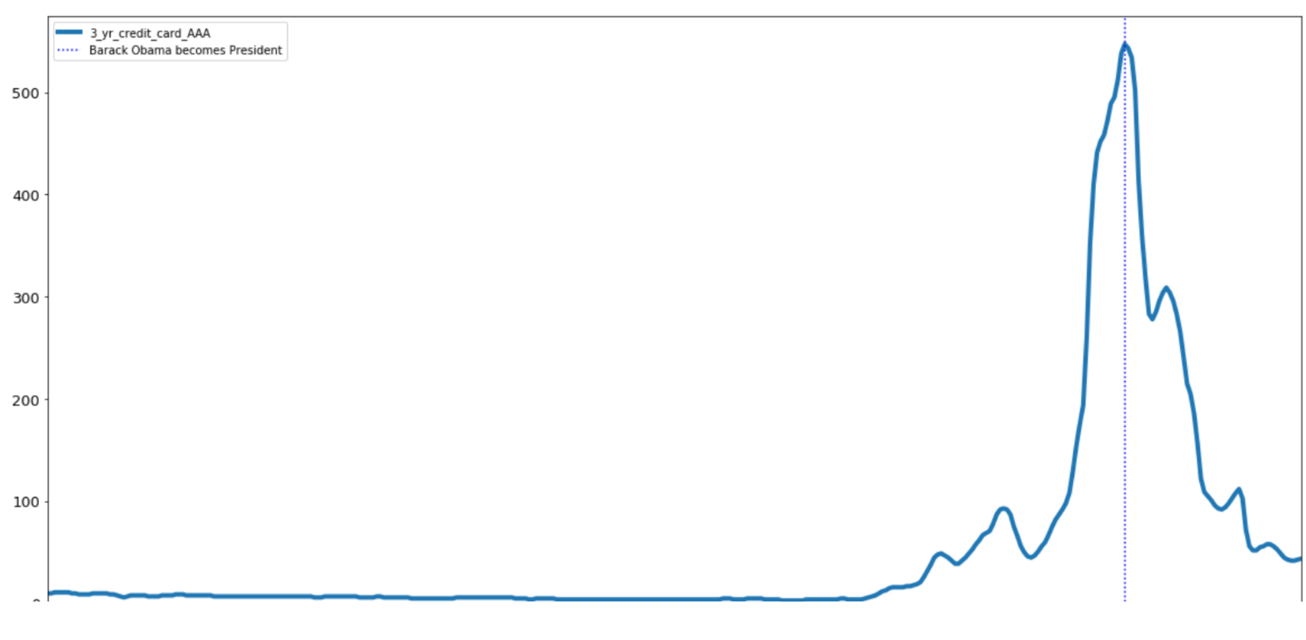
Figure 6 shows the growth of ABS outstanding from 1995 to late 2008. We can clearly see that "Other" and "Home Equity" represented more than 2/3 of the total market.

The “Other” category included derivative instruments, such as ABS of ABS, CDOs, CDOs squared, etc. Once you had problems in a particular asset class, the ABS that included that asset class had problems, as well as the ABS of ABS that contained that asset.

Figure 7 below shows that the spread for a typical 3-year credit card AAA rated securitization increased from about 0.2% over LIBOR in early 2007, to almost 6% over LIBOR in December 2009. Similarly, the volume of securitizations in the market decreased from over USD $1.2 trillion in 2006, to less than USD $30 billion in the first quarter of 2009, a 98% drop with respect to its peak.

> Problems in two US financial institutions contributed to the increase of spreads in ABS: 1) Bears Stearns, the fifth largest US bank, and 2) Lehman Brothers, the 4th largest bank. Both had proprietary capital in ABS, in addition to being heavily involved in the creation of all types of securitization products worldwide

This drastic decrease in the available funds for financing and its associated increase in the cost of financing created a global financial gridlock that needed to be handled by the authorities.



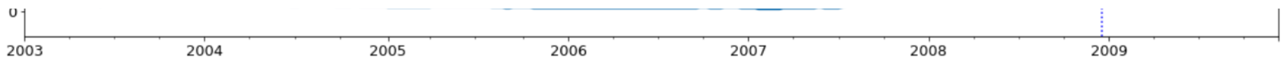



**Figure 7**: Spreads for AAA rated credit card ABS (as well as all other ABS) skyrocketed in late 2008 and peaked as Barack Obama became President of the US. Issuance of credit card ABS were fueling the domestic and global economies. Source: SAGA Capital, LLC

In order to jump start the domestic economy and try to reverse the pervasive trends it was experiencing, two new plans were put into place: **1) the TARP or Troubled Assets Relief program**, implemented in the last days of the Bush administration, and **2) the TALF or Term Asset Loan Facility program**, announced by the Obama administration a few days after President Obama took office.

> TARP was in essence a bailout of everybody but Lehman Brothers, designed by the Bush administration. TALF was a reactivation program of the US economy, designed by the Obama administration, but it had a built-in catch 22 for the institutions bailed out by TARP.

## The Federal Reserve jumpstarts the economy

On December 15, 2008, Barack Obama was sworn in as President of the US. On February 2, 2009, the US Federal Reserve announced the TALF's Master Loan and Security Agreement, outlining the rules of the program.

Despite the massive injection of capital into the US banking system in the last days of the Bush administration, there was no increase in consumer lending or spending.

The US Federal Reserve and the Department of the Treasury under the Obama administration recognized that reviving the ABS market was critical to restoring the flow of credit to consumers. Accordingly, the Fed and the Treasury launched the TALF program — a USD $1 trillion program designed to entice investors into buying legacy and newly issued AAA rated securitizations backed by specific asset classes.

By providing most of the financing for investors' purchases, the Fed and Treasury hoped that renewed demand for ABS would drive down credit spreads for the issuing banks and finance companies and encourage investors to buy, promoting further securitization of loans. In turn, this should make more loans available to more consumers at lower rates.

The TALF program provided up to 95% of the funds required to buy a qualified bond, at rates (fixed or floating) of LIBOR plus 1% over 1-3 years, with a minimum loan amount of USD $10 million.

The TALF program enhanced the risk-reward profile for investors by providing true non-recourse loans. As a result, substantial leverage was possible and thus raised potential returns significantly. Losses were not leveraged, as these were true non-recourse loans that did not require posting of additional collateral or forced redemption of collateral.

To initiate the process, a US investment company (either a newly formed or existing one) that met US government requirements had to open an account with a Primary Dealer and identify eligible bonds that the investor wanted to buy.

## TALF purchase walk-through

To understand the logistics of the program back then, let's walk through a hypothetical purchase.

First, let's assume that we are back in the first week of the announcement of the TALF program in early 2009. Let's assume that a hypothetical fixed income Portfolio Manager (PM) at Goldman Sachs checks his Bloomberg screen and sees something like the table below, under the first page of thousands of TALF eligible securities:

| Security | Type | Rating | Term | Collat | Orig(M) | Lead Mgr | Cusip |
|---|---|---|---|---|---|---|---|
| SBAP 2009-20D 1 | ABS | AAA | 3 | ASSET | 336,856 | BOA, CS | 83162CSL8 |
| WOART 2009-A A1 | ABS | AAA | 3 | AUTO | 163,000 | BCG, BOA | 98156YAA9 |
| WOART 2009-A A2 | ABS | AAA | 3 | AUTO | 192,000 | BCG, BOA | 98156YAB7 |
| WOART 2009-A A3 | ABS | AAA | 3 | AUTO | 248,000 | BCG, BOA | 98156YAD3 |
| WOART 2009-A A4 | ABS | AAA | 3 | AUTO | 147,000 | BCG, BOA | 98156YAF8 |
| WOART 2009-A B | ABS | AAA | 3 | AUTO | 78,389 | BCG, BOA | 98156YAH4 |
| CABMT 2009-1A A | ABS | AAA | 3 | CARD | 425,000 | WS, JPM | 126802BD8 |
| CARMX 2009-1 A1 | ABS | AAA | 3 | AUTO | 182,000 | BOA, JPM | 14312WAA1 |
| CARMX 2009-1 A2 | ABS | AAA | 3 | AUTO | 254,000 | BOA, JPM | 14312WAB9 |
| CARMX 2009-1 A3 | ABS | AAA | 3 | AUTO | 260,000 | BOA, JPM | 14312WAD5 |
| CARMX 2009-1 A4 | ABS | AAA | 3 | AUTO | 144,000 | BOA, JPM | 14312WAF0 |
| CABMT 2009-1X A | ABS | AAA | 3 | CARD | 425,000 | WS, JPM | BCC16XHG6 |
| WFNMT 2009-A A | ABS | AAA | 3 | CARD | 560,000 | BCG, JPM | 981464BQ2 |
| NAROT 2009-A A1 | ABS | AAA | 3 | AUTO | 357,000 | BOA, JPM | 65476AAA3 |
| NAROT 2009-A A2 | ABS | AAA | 3 | AUTO | 323,000 | BOA, JPM | 65476AAB1 |
| NAROT 2009-A A3 | ABS | AAA | 3 | AUTO | 493,000 | BOA, JPM | 65476AAD7 |
| NAROT 2009-A A4 | ABS | AAA | 3 | AUTO | 196,522 | BOA, JPM | 65476AAF2 |
| NAROT 2009-A CTFS | ABS | AAA | 3 | AUTO | 115,860 | BOA, JPM | 65476AAH8 |
| CCCIT 2009-A1 A1 | ABS | AAA | 3 | CARD | 3,000,000 | CITG | 17305EEM3 |

A few of TALF eligible securities. Source: SAGA Capital

The Goldman Sachs PM decides that the CCCIT 2009-A1 A1 AAA securitization of credit cards loans issued by Citigroup in the amount of USD $3 billion is a good bond to buy. Checking LIBOR rates and spreads at that time he might have seen:

| | libor | 3_yr_auto_AAA | 3_yr_student_loan_AAA | 3_yr_helc_AAA | 3_yr_credit_card_AAA |
|---|---|---|---|---|---|
| date | | | | | |
| 2009-01-16 | 178 | 406 | 271 | 1546 | 413 |
| 2009-01-23 | 187 | 321 | 247 | 1478 | 359 |
| 2009-01-30 | 194 | 306 | 228 | 1363 | 318 |
| 2009-02-06 | 207 | 327 | 166 | 1300 | 283 |
| 2009-02-13 | 201 | 334 | 146 | 1312 | 278 |

ABS spreads

The Goldman Sachs PM decides to buy USD $100MM worth of that bond, expecting to make approximately **4.79%** (278bps + 201bps ) annual return for 3 years on his USD $100MM investment. Not bad for a fixed income portfolio manager.

What about a hedge fund manager? 4.79% annual return for 3 years is not the kind of return a hedge fund is looking for, especially when hurdle rates greater than 5% are not uncommon.

However, TALF leverage ***completely changed the distribution of risk***, and made the low yield AAA ABS security the best game, but also the least known game in Wall Street.

$$Annual\,Return_{(Asset,Term)} = \frac{((LIBOR + Spread) * fRiskCapital) + ((Spread - Haircut) * (1 - fRiskCapital))}{fRiskCapital}$$

After checking all the documentation, requirements, etc., the expected annual returns under no loss scenario when buying TALF eligible securities could be summarized in the above formula.

Continuing with our hypothetical Goldman Sachs PM, he asks his Jr. Analyst to get a current list of TALF leverage and finds out that for a 3 year credit card AAA ABS, the Fed via TALF can lend 94 cents for every 6 cents he commits, at an interest rate of LIBOR minus 100 basis points, with a minimum investment of $10MM USD.

| Sector | Term 1 | 2 | 3 |
|---|---|---|---|
| Auto | 10% | 11% | 12% |
| Credit Card | 5% | 5% | 6% |
| Student Loan | 8% | 9% | 10% |
| HELOC | 12% | 13% | 14% |

Actual TALF leverage for eligible securities in February 2009

Applying the formula above, the PM sees that the expected annual return of acquiring that bond (in a zero default scenario) changed from **4.79% to 32.67%** for the next 3 years!

```
((((2.01% + 2.78%)*0.06) + ((2.78%-1.00%)*(0.94)))/0.06) = 32.67%
```

Now, this sort of return is interesting even for the long-short hedge funds that don't even look at ABS. But, if the hedge fund had survived the crisis, the last thing they would want to do is lock funds in a 3 year buy and hold strategy.

Additionally, the quants who model stock market movements in long-short hedge funds and/or do high frequency algorithmic trading tend not to have much practical knowledge about securitization tranches, credit migration models, binomial expansion models, copulas, etc. They were lost and happy they still had job. " No, thank you, I pass" was probably the thinking in their minds to the opportunity in the slight chance they came across it.

The best part of the TALF program for investment managers was that, again, the leveraged capital was a non-recourse loan, meaning that in case of default, the loan from the Federal Reserve did not have to be paid back!

So, maybe the best candidate to take advantage of this opportunity was our hypothetical PM from Goldman Sachs (or Bank of America, or JP Morgan, or any other.) After all, they have the knowledge and the funds, right?

Wrong. Since Goldman Sachs as well as a large chunk of the banking sector had been bailed out directly or indirectly by Bush's TARP program, they were precluded from participating in Obama's TALF program. In case you did not know, Bush's bailout program for Goldman Sachs (via AIG bailout) was masterminded by the then Secretary of the Treasury and former CEO of Goldman Sachs, Henry "Hank" Paulson. (Years earlier I had worked for AIG Risk Finance, so I have good knowledge of what was going on).

So, effectively, this opportunity was initially taken advantage of by a few high net worth individuals, a small number of hedge funds and family offices that understood ABS and were willing to take risks, and an even smaller number of pools of quants (or data scientists, although they were not called that back then, but "special opportunities quants" or "risk finance quants") that understood the models, had domain expertise, and who pooled their resources to meet the investment requirements.

In the institutional side, PIMCO and BlackRock are two largest asset managers that did take advantage of the opportunity big-time (later on in the game), but that's another story.

## Expected returns of a systematic TALF purchase program- Monte Carlo Simulation using Python, Pandas, Numpy, SciPy & Statsmodels Time Series Analysis

So we are back in 2009 and let's picture the following scenario: I am a quant / data scientist advising a family office that had never invested in low yield ABS.

The investment manager at the family office wants to know why I think investing $100MM in TALF qualified securities is a good idea. To decide if they will commit funds to my idea, my model needs to answer the following questions:

- What is the distribution of expected annual returns if the family office commits $100MM in 4 weeks? I recommended $40MM the first week, $30MM the 2nd week, $20MM the 3rd week, and $10MM by the 4th week.
- What is the probability of losing money if they commit to buying every bond they can with the $100MM of capital, regardless of the spread at purchase time?
- How many Fallen Angels can we expect in the TALF eligible securities?
- What is the probability of making more than a 5% hurdle rate, if we discriminate which bonds to buy based on their spreads and leverage at the moment of purchase?

- What is the median expected annual return of that strategy, assuming none of the bonds default?
- What are the expected defaults in the proposed portfolio in basis points?
- What kind of average leverage can the family office expect?
- What is the average buy and hold period of the portfolio of ABS?

There are some last questions I will answer in this article that refer to the performance of my model in real-life:

- How well did my interest rate model fit reality?
- How well did my modeled correlations behave?

Before answering these questions, I will say that the model behaved as expected and the returns were realized, as you will learn in the rest of this article. This was not an academic exercise but a real trade, and yes, hundreds of billions of USD were available to borrow for 3 years at rates of LIBOR minus 1%.

## Modelling

To keep things short in this article, I'll assume the 1 and 2 year LIBOR rates are the same as the 3 year LIBOR rate and base all my calculations on the modeled 3 year LIBOR rate (conservative assumption, since the 3 year rate is generally more expensive than the 1 or 2 year rate).

The one year ABS spread will be the 3 year ABS spread minus the average historical difference (up to February 18, 2009) between those 2 terms. The two year ABS spread will be halfway that difference.

These are the steps for the whole analysis:

1. Simulate N paths of correlated 3-year AAA spreads for credit cards, student loans, home equity lines of credit (HELOC), car loans, and LIBOR rates, sampled weekly and starting at the date of the announcement of the TALF Master Loan Agreement.
2. Recreate the 1 and 2 year spreads from the 3 year spreads.

3. For every week of the 4 week purchase program, simulate purchases of one of the 4 types of qualified assets, with probabilities given by the distribution of ABS outstanding as of January 2009. Apply Fed haircuts to corresponding assets, and use spreads for that asset class calculated in 1 and 2 above to estimate the no loss expected return.

4. Simulate migration of the bond from its initial AAA rating to other ratings and the state of default "D". Keep track of the bond's credit quality migration over its life.

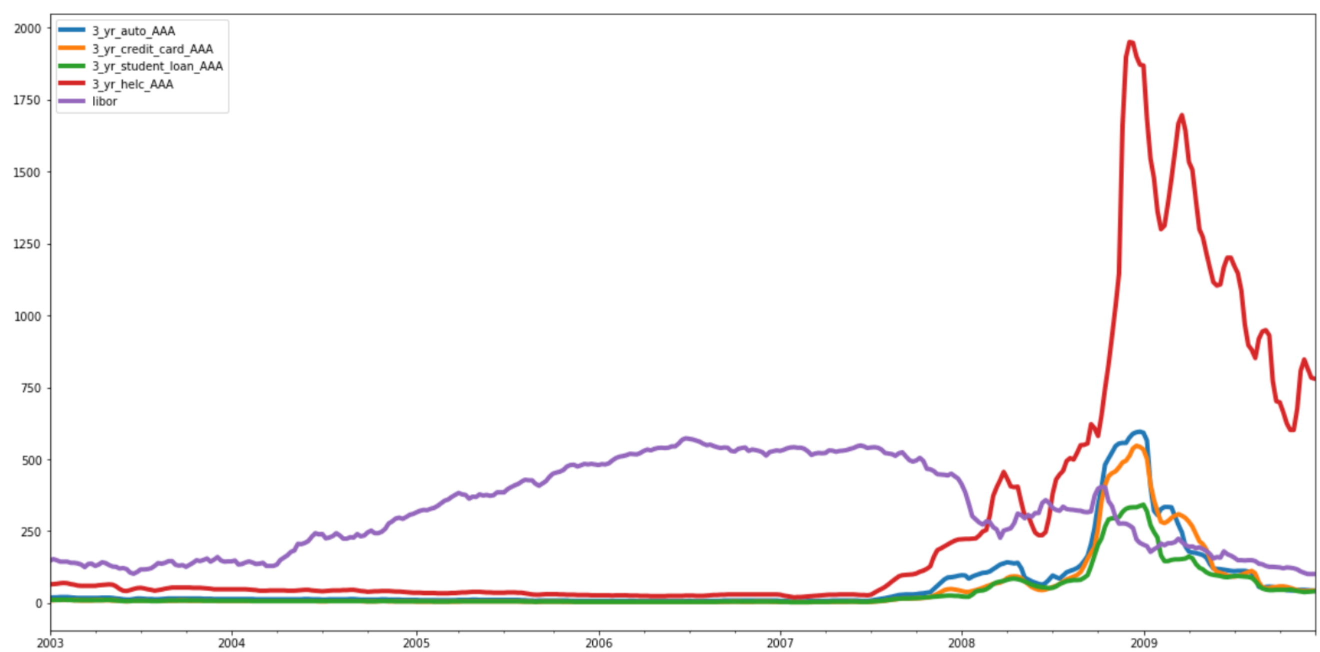
5. ABS is not traded, but held by the Fed as collateral for the loan until maturity or bond default.

6. Strategy is buy and hold, so once an asset is acquired in the simulation, we don't need to model market price fluctuations. No mark to market.

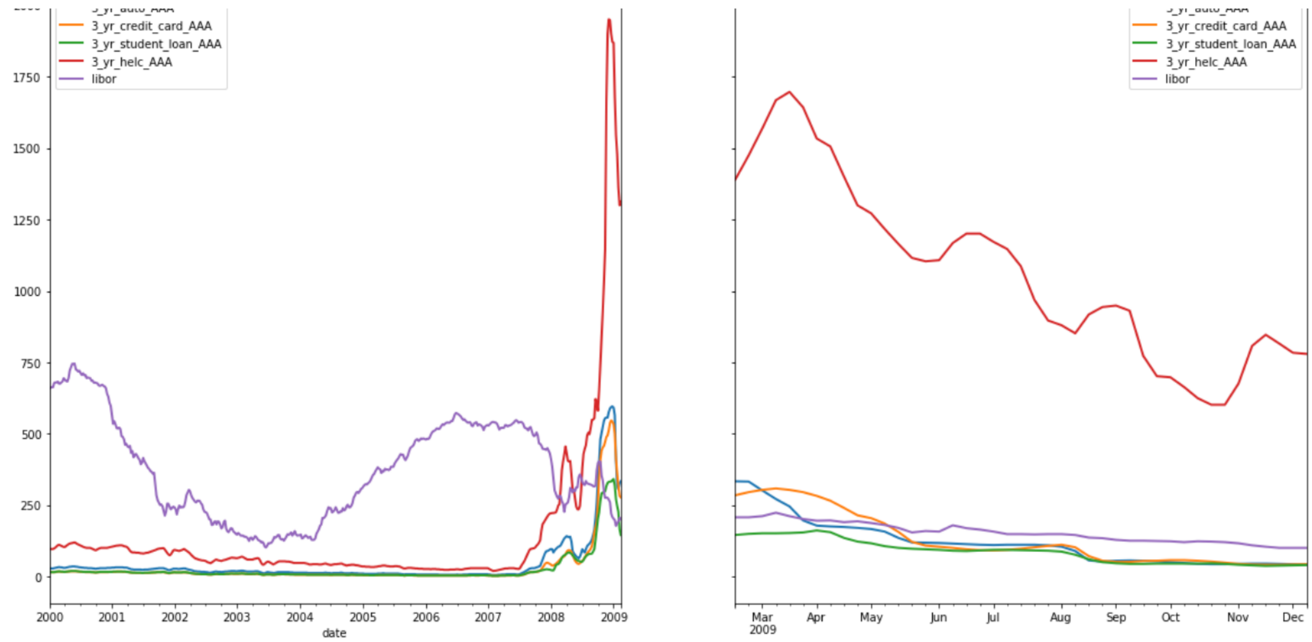
7. Show relevant distributions to answer questions.

## Train, Test, Split

The 2000–2010 time series looks like this:



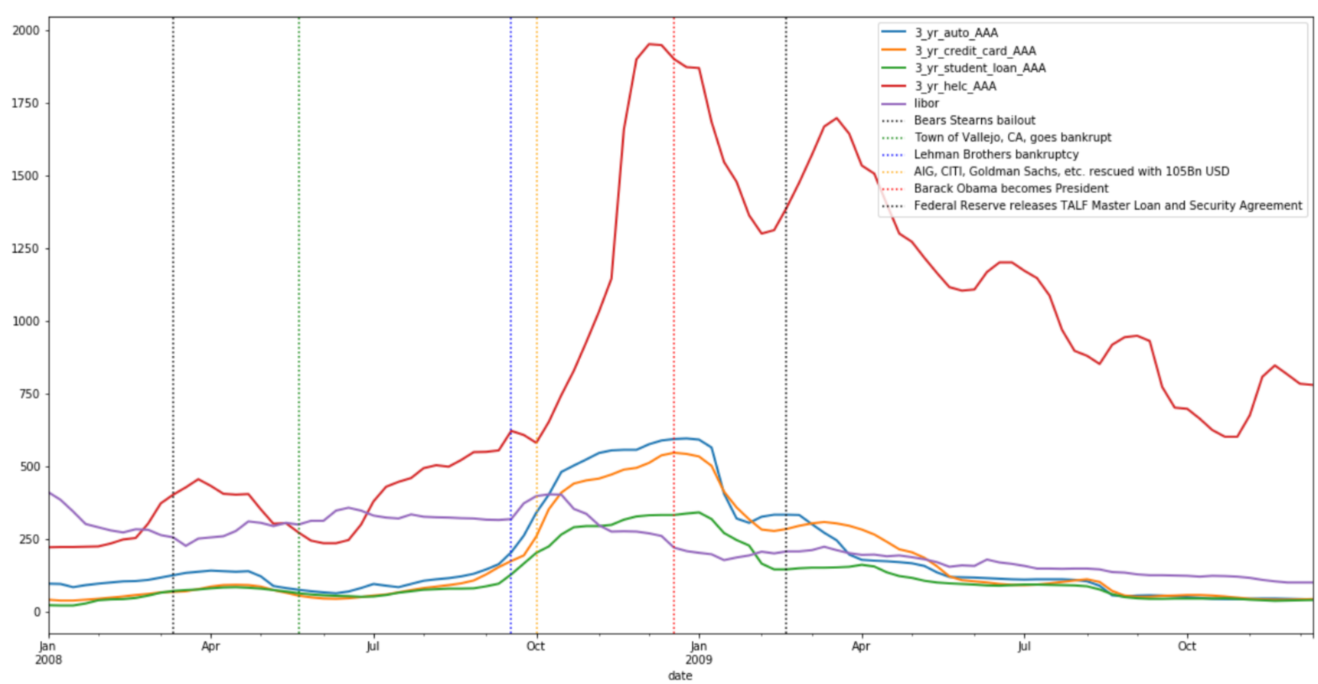

I split the data in 2 segments: before TALF announcement (for training) and after TALF announcement. The train data is what I had available at the moment of TALF announcement.




The crisis period is the one below. The period for testing starts in the last black dotted line, shortly after the spreads started to come down from their peak, when President Obama took office.

## Some historical milestones to determine train, test, split

```
In [4]: xcoords = ['20080314', '20080517','20080915', '20081003', '20081215', '20090218']
        colors = ['k','g','b','orange','r', 'black']
        legend = ['Bears Stearns bailout',
                  'Town of Vallejo, CA, goes bankrupt',
                  'Lehman Brothers bankruptcy',
                  'AIG, CITI, Goldman Sachs, etc. rescued with 105Bn USD',
                  'Barack Obama becomes President',
                  'Federal Reserve releases TALF Master Loan and Security Agreement']
```




Milestones of the financial crisis

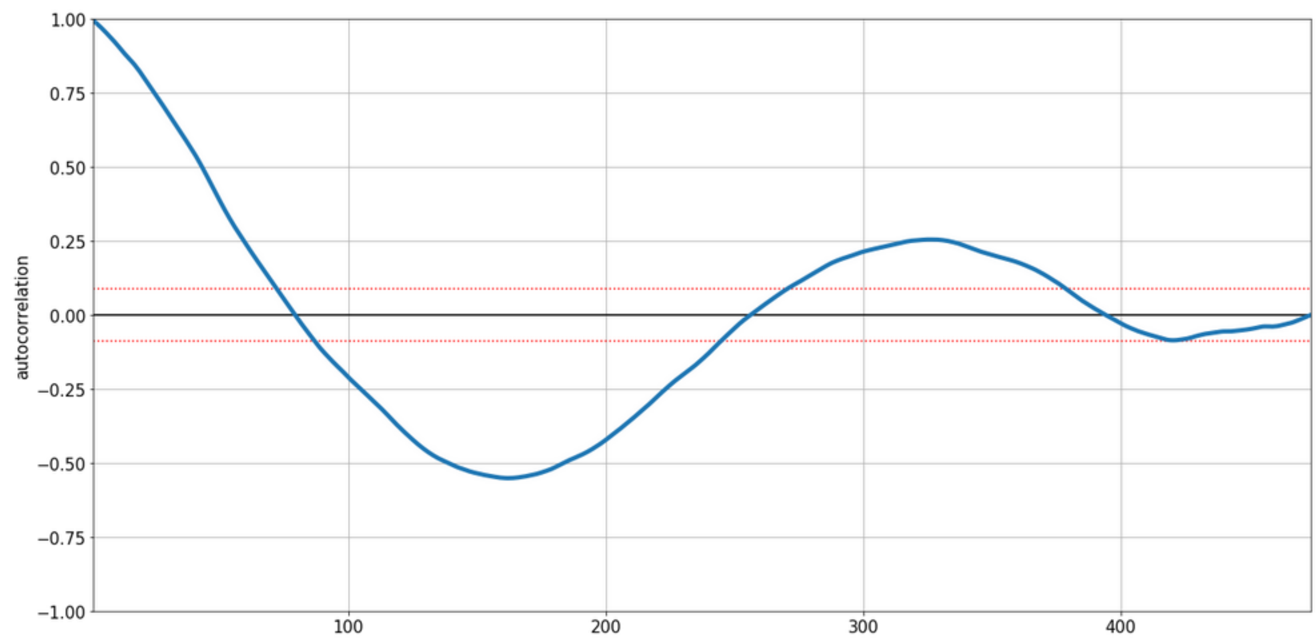
## Mean reversion and auto-correlation

Mean reversion is the assumption that an asset's price will tend to move to some long term average. This might happen with interest rates, commodities, currencies, etc. Deviations from average levels are expected to revert to the average.

Auto-correlation or serial correlation is the correlation of a signal with itself as a function of a time lag. It is often used in signal processing for analyzing functions or series of values.

Below are the auto-correlation plots for LIBOR and 3-year spreads for AAA securitization of auto loans, credit card loans, student loans, and HELC loans.

```
#LIBOR
plt.figure(figsize=(20,10))
df_libor, lags_libor, ax_libor = autocorrelation_and_significance(df_weekly_train['libor'], linewidth=4)
```




Auto correlation for LIBOR rate

By inspecting the lags for the LIBOR rate, we can see there are several lags that are statistically significant with positive and negative auto-correlation. All of them are good candidates to predict future LIBOR rates based on many lagged parameters of itself. I decided to use a simple auto regressive model, since more complex models that model shocks could show distortions since we were already in a shock scenario. Another model I could have used is the Vasicek model, which incorporates factors such as the speed of reversion.

With the simple auto-regressive model I decided to use, today's LIBOR rate can be set equal to some mean value, plus a fraction of yesterday's value (phi), plus some noise. For a process to be stationary, the phi values needs to be between -1 and +1.

If the process is just noise, phi=0, and if it is a **random walk**, phi=1. If phi is negative, the process shows **mean reversion**. If phi is positive, the process shows **momentum**.

In the chart below, Rt is a time series of interest rates. Using the auto-correlation data of the LIBOR rate above, we could define future rates as a function of its lags. The first one is AR1, the second one is AR2, etc.

AR(1)

$$R_t = \mu + \phi_1 R_{t-1} + \epsilon_t$$

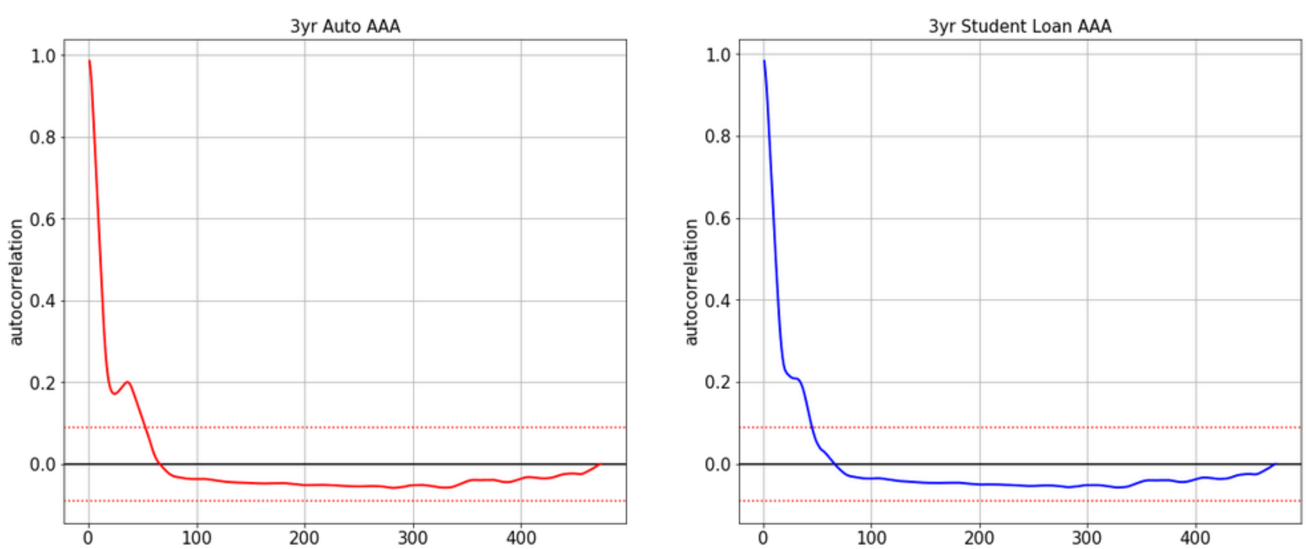
AR(2)

$$R_t = \mu + \phi_1 R_{t-1} + \phi_2 R_{t-2} + \epsilon_t$$

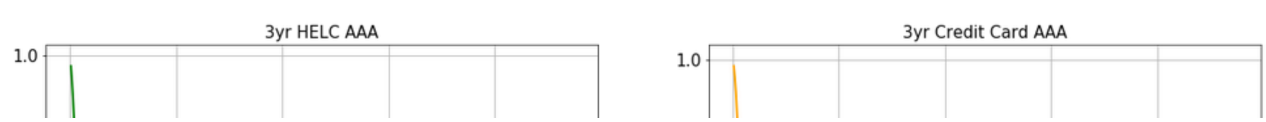
AR(3)

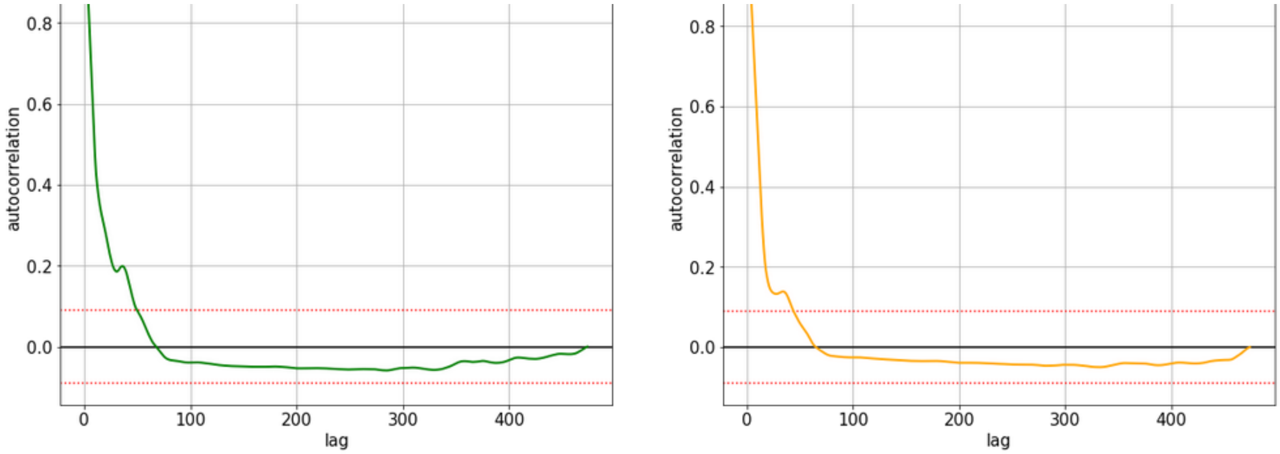
$$R_t = \mu + \phi_1 R_{t-1} + \phi_2 R_{t-2} + \phi_3 R_{t-3} + \epsilon_t$$

Below are the auto correlation plots for ABS spreads:




Auto-correlation for auto loans and student loans time series






Auto-correlation for HELC and credit cards time series

The code to generate the auto-correlation and significance plot above is in the gist below, extending pandas auto-correlation functionality a little further in order to return a data frame as well as list of lags at a significance of ~1.96 standard deviations:

```
def autocorrelation_and_significance(series, ax=None, **kwds):
    """Autocorrelation and significant lags for time series."""
    n = len(series)
    data = np.asarray(series)
    if ax is None:
        ax = plt.gca(xlim=(1, n), ylim=(-1.0, 1.0))
    plt.rc('xtick', labelsize=15)
    plt.rc('ytick', labelsize=15)
    mean = np.mean(data)
    c0 = np.sum((data - mean) ** 2) / float(n)

    def r(h):
        return ((data[:n - h] - mean) *
                (data[h:] - mean)).sum() / float(n) / c0
    x = np.arange(n) + 1
    y = lmap(r, x)
    z95 = 1.959963984540054
    z95l = -z95/np.sqrt(n)
    z95h = z95/np.sqrt(n)
    ax.axhline(y=z95h, linestyle=':', color='red', )
    ax.axhline(y=0.0, color='black')
    ax.axhline(y=z95l, linestyle=':', color='red')
    ax.set_xlabel("lag", fontsize=15)
    ax.set_ylabel("autocorrelation", fontsize=15)
    ax.plot(x, y, **kwds)
    if 'label' in kwds:
        ax.legend()
    ax.grid()
```

```
    df = pd.DataFrame({'autocorrelation': y,
                       'lag': x})
    lags = np.sort(np.append(df[df['autocorrelation'] > z95h]['lag'],
                             df[df['autocorrelation'] < z95l]['lag']))
return df, lags, ax
```

autocorr.py hosted with ♡ by GitHub view raw

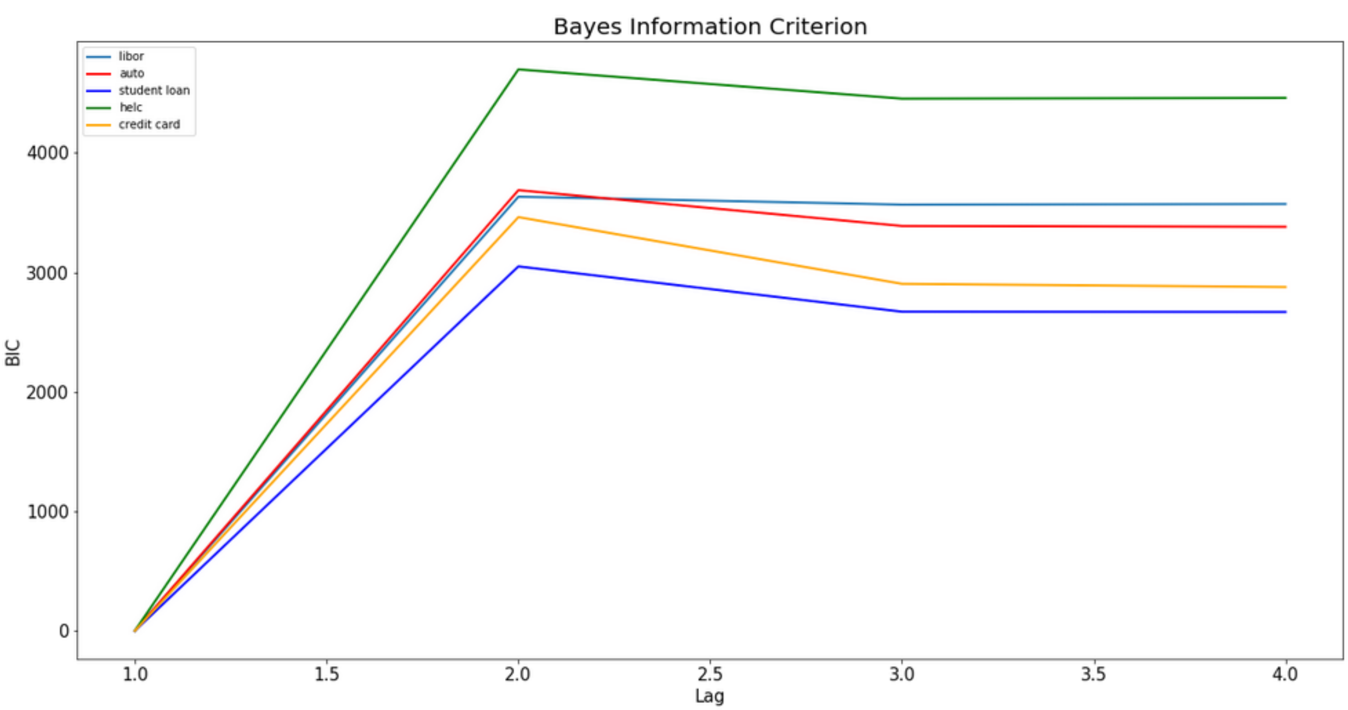
Auto-correlation and significant lags for time series

By inspecting the lags for ABS and LIBOR rates, we see there are many lags we could use, but how do we avoid over fitting or under fitting our time series model?

## Order of auto-regressive models

Although we could use all valid lags for all assets, we will probably be over fitting our models.

A solution is fitting many models (AR1, AR2, AR3, etc.), measuring the Akaike Information Criterion (AIC) and/or the Bayesian Information Criterion (BIC) for each model, and determining which order of AR gives us the lowest BIC or AIC.




BIC for LIBOR and selected AAA ABS spreads

According to their AIC or BIC, models higher than AR1 might contribute to over fitting our model, therefore, I will simulate rates using AR1 models for all assets.

The code to determine the optimal parameters is below:

```python
def optimal_params_ar_model(data, lags_to_test, cap=4, test_criteria='BIC', **kwds):
    """Optimal lags using Bayes or Akaike Information Criteria.
    Given a time series and significant lags returned by the
    autocorrelation_and_significance function, this function tests n values (cap)
    to find out if auto regresive models of order > 1 are worth exploring.
    Test criteria can be Akaike ('AIC') or Bayes ('BIC').
    """
    ax = plt.gca()
    plt.rc('xtick', labelsize=13)
    plt.rc('ytick', labelsize=13)
    num_lags = len(lags_to_test)
    information_criteria = np.zeros(num_lags)
    for lag in list(lags_to_test)[:cap]:
        mod = ARMA(data.values, order=(lag, 0))
        res = mod.fit()
        if test_criteria == 'BIC':
            information_criteria[lag] = res.bic
            ax.set_title('Bayes Information Criterion', fontsize=20)
            ax.set_ylabel('BIC', fontsize=15)
        elif test_criteria == 'AIC':
            information_criteria[lag] = res.aic
            ax.set_title('Akaike Information Criterion', fontsize=20)
            ax.set_ylabel('AIC', fontsize=15)

    ax.set_xlabel('Lag', fontsize=15)
    ax.plot(lags_to_test[:cap], information_criteria[:cap], **kwds)
    ax.legend(loc='best')

    return ax
```

optimal_params_ar.py hosted with ♡ by GitHub view raw

Once I determined the optimal lags to use, I created a dictionary of AR parameters for all the ABS spreads and the LIBOR rate.

```python
def ar_param_dictionary(train_df, order):
    """Parameters of autoregressive models.

    Given a train df, this function fits auto regressive models of given order
    for the different time series in the train df and stores a summary
    of results, the AR1 value, and the volatility (standard deviation) of
    observations in a python dictionary.
    """
```

```python
    ar_params = {}
    for i, col in enumerate(train_df):
        asset = train_df[col]
        mod = ARMA(asset, order=(order, 0))
        res = mod.fit()
        ar_params[i] = {'name': col,
                        'summary': res.summary(),
                        'AR1': res.arparams[0],
                        'vol': res.sigma2}
    return ar_params
```

ar_params_dictionary.py hosted with ♡ by GitHub view raw

## Correlations

The MC simulation needs to generate simulated rates that keep the properties of their historical correlations.

For simplicity's sake, I calculated the correlation among the different asset classes at the same maturity, and assumed that the correlation of, for example, the 3 year auto ABS AAA spread and 1 year auto AAA ABS spread was 100% (which is close to the behavior in real life most of the time).

For the multivariate random number generator, I initially used the scipy function here, but found that using a Cholesky decomposition was computationally more effective.

I will also generate a date_index that will contain all the future dates of our analysis up to the potential longest maturity of the simulated bonds we will buy.

The code is below:

```python
sims = 100000
longest_tenor_yrs = 3
purchase_weeks = 5
date_ix = pd.date_range(start=df_weekly_train.index[-1],
                        periods= (purchase_weeks + 1) + longest_tenor_yrs * 52, freq='7D')
```

```python
all_sims, assets_sims = pyabs.simulate_several_sets_correlated_rates(df_weekly_train, sims, date_ix, ar_params_dict)
```

```
100%|██████████| 100000/100000 [03:17<00:00, 505.90it/s]
```

```python
def simulate_correlated_random_numbers(corr_matrix, n=1000):
    """Multivariate random normal.

    A generalization of the one-dimensional normal distribution to higher
    dimensions, using Cholesky decomposition.
```

```
    dimensions, using Cholesky decomposition.
    https://math.stackexchange.com/questions/2079137/generating-multivariate-normal-samples-why-
    """
    upper_cholesky = cholesky(corr_matrix)
    rnd_numbers = np.random.normal(0.0, 1.0, size=(n, corr_matrix.shape[0]))
    ans = rnd_numbers@upper_cholesky
    return(ans)


def simulate_single_set_interest_rates(train_df, date_ix, ar_params_dict, vol_stress=1):
    """Simulate 1 path of multiple future interest rates.

    Given an historical time series of interest rates, an index of future dates,
    and a dictionary of autoregressive parameters, this function generates a
    path of correlated interest rates.
    """
    corr_matrix = train_df.corr().as_matrix()
    fut_rates = {}
    for i in range(train_df.shape[1]):
        fut_rates[i] = np.zeros(len(date_ix))
        fut_rates[i][0] = train_df.iloc[-1, i]

    corr_rnd = simulate_correlated_random_numbers(corr_matrix)
    for k in range(train_df.shape[1]):
        for z in range(len(date_ix)):
            # skip the first value, since it is the seed value for the sim
            if z != 0:
                fut_rates[k][z] = fut_rates[k][z-1]*ar_params_dict[k]['AR1'] + ar_params_dict[k]
                fut_rates[k][z] = (fut_rates[k][z])*(vol_stress)
                # for cases ofa simulatd negative spread, set the sread to the
                # previous positive spread
                if fut_rates[k][z] < 0:
                    fut_rates[k][z] = fut_rates[k][z-1]
    rates_df = pd.DataFrame(fut_rates, index=date_ix)
    rates_df.columns = train_df.columns
    return rates_df
```

correlated_assets_movements.py hosted with ♡ by GitHub view raw

## Probabilities of AAA rated assets being downgraded or going to default in the holding period

To estimate this, I created a function that defines a transition matrix based on generic historical transition probabilities for AAA ABS in one year, excluding mortgages.

This 1 year transition matrix can be used to simulate transitions from any initial state (AAA, AA, A, BBB, BB, B, CCC) to any other state plus “D” using a one state Markov process.

We can then estimate the probabilities of transition over a given number of years by adding the 1-year transition matrix to a recursive function with time as a parameter. With this, we can derive many transition vectors running the function in a simulation, which will help us estimate probabilities.

```
pyabs.estimate_1yr_transition('AAA')
```

```
'AAA'
```

```
pyabs.estimate_transition_vector('B', 5)
```

```
['B', 'BB', 'BB', 'BB', 'BB']
```

pyabs, credit migration estimation

```
def estimate_1yr_transition(initial_rating='AAA'):
    """Simulate 1 period rating transition.

    This function estimates the transition from any given rating
    'AAA', 'AA', 'A', 'BBB', 'BB', 'B', 'CCC' to the same rating plus the 'D'
    (default) state, based on observed trasitions for ABS in 1 year, excluding
    mortgages. These are approximations, and each asset class should have its
    own transition matrix.
    """
    data = np.array([[9.081e-01, 8.330e-02, 6.500e-03, 9.000e-04, 6.000e-04, 3.000e-04, 2.000e-0
                     [7.000e-03, 9.065e-01, 7.790e-02, 6.400e-03, 6.000e-04, 1.300e-03, 2.000e-0
                     [9.000e-04, 2.270e-02, 9.105e-01, 5.520e-02, 7.400e-03, 2.600e-03, 1.000e-0
                     [2.000e-04, 3.300e-03, 5.950e-02, 8.693e-01, 5.300e-02, 1.170e-02, 1.200e-0
                     [3.000e-04, 1.400e-03, 6.700e-03, 7.730e-02, 8.053e-01, 8.840e-02, 1.000e-0
                     [0.000e+00, 1.100e-03, 2.400e-03, 4.300e-03, 6.480e-02, 8.346e-01, 4.070e-0
                     [2.200e-03, 2.200e-03, 2.200e-03, 1.300e-02, 2.380e-02, 1.124e-01, 6.486e-0
    initial_ratings = ['AAA', 'AA', 'A', 'BBB', 'BB', 'B', 'CCC']
    transition_to = initial_ratings.copy()
    transition_to.append('D')
```

```python
    p_transition = pd.DataFrame(data, index=initial_ratings, columns=transition_to).transpose().
    final = np.random.choice(list(p_transition[initial_rating].keys()), 1, p=list(p_transition[i
    return final


def estimate_transition_vector(initial_rating, years):
    """Simulate rates upgrades or downgrades in n years.

    This function simulates the movement of the initial rating over time,
    by using the function recursively, i.e.: the initial rating feeds the
    function, and the output of the function feeds the function again until
    n periods have been completed. This is a one state Markov process.
    """
    input_list = [initial_rating]
    if input_list == []:
        return 0
    else:
        for i in range(years-1):
            new_rating = estimate_1yr_transition(initial_rating=input_list[-1])
            input_list.append(new_rating)
            if new_rating == 'D':
                return input_list
    return input_list
```

markov_transition.py hosted with ♡ by GitHub view raw

## Simulated leveraged purchases on a single interest rate scenario

The process here is as follows: With one of the simulated and properly correlated LIBOR rate and spread scenarios, we simulate the purchase of any of the 4 eligible assets classes in week one, at whatever prevalent market rates the simulation shows, and with the leverage provided by the Federal Reserve.

```python
f_risk_capital = {'auto_AAA':         {1: 0.10,
                                        2: 0.11,
                                        3: 0.12},
                  'student_loan_AAA': {1: 0.08,
                                        2: 0.09,
                                        3: 0.10},
                  'helc_AAA':         {1: 0.12,
                                        2: 0.13,
                                        3: 0.14},
                  'credit_card_AAA':  {1: 0.05,
                                        2: 0.05,
                                        3: 0.06}}
```

```python
p_term = {1: 0.20,
          2: 0.30,
          3: 0.50}
```

```python
p_issuance = {'auto_AAA':         0.20,
              'student_loan_AAA': 0.20,
              'helc_AAA':         0.20,
              'credit_card_AAA':  0.40}
```

TALF leverage per type of ABS as a function of term (f_risk_capital), probabilities of 1, 2, and 3 years issuance (p_term), and probabilities of acquiring each of the asset classes (p_issuance)

With those simulated market conditions, we can estimate the no loss return, and decide if we buy or not. However, we will not know if we would incur losses in any of those purchases. To know that, we need to model defaults, but we will not use that knowledge for our decision to buy. We will only use expected annual return and hurdle rate.

| | asset | purchase_week | purchase_date | maturity_date | term | f_risk_capital | risk_capital | fed_loan | total_purchase | libor | spread_over_libor | exp_annual_r | final_rating |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 3_yr_auto_AAA | 1 | 2009-02-20 | 2012-02-17 | 3 | 0.12 | 30 | 220 | 250 | 47 | 387 | 0.254218 | AA |
| 1 | 1_yr_student_loan_AAA | 2 | 2009-02-27 | 2010-02-26 | 1 | 0.08 | 20 | 230 | 250 | 47 | 162 | 0.093173 | AAA |
| 2 | 3_yr_auto_AAA | 3 | 2009-03-06 | 2012-03-02 | 3 | 0.12 | 20 | 146 | 166 | 74 | 173 | 0.078244 | AAA |
| 3 | 1_yr_helc_AAA | 4 | 2009-03-13 | 2010-03-12 | 1 | 0.12 | 20 | 146 | 166 | 38 | 840 | 0.630914 | AAA |
| 4 | 1_yr_auto_AAA | 5 | 2009-03-20 | 2010-03-19 | 1 | 0.10 | 10 | 90 | 100 | 38 | 290 | 0.204688 | AAA |

Example of a run of the simulated purchase program under one of the simulated and properly correlated interest rate and spread scenarios

In the simulated scenario above, in week 3 (index 2) on simulated March 6, 2009 we could have bought USD $166 million of a AAA auto loan ABS maturing on March 2, 2012, committing only USD $20MM of our capital and USD $146MM from a loan from the Fed. With those prevalent market conditions, our expected annual return on that purchase (no loss scenario) is 7.82%.

Assuming a hurdle rate of 10%, we would not have bought that specific bond. Additionally, our simulated 3-year Markov transition shows that bond kept its AAA rating during its life. So, in this particular scenario, we would have purchased bonds at index 0, 3, and 4. Those simulated market conditions would have allowed us to lock USD $250MM @ 25.42% for 3 years, USD $166MM @ 63.09% for 1 year, and USD $100MM @ 20.46% for 1 year. That turns out to be USD $516MM in investments with only USD $60MM of capital.

This particular simulation shows that the first bond suffered a simulated downgrade from AAA to AA, which could have slightly affected its market price at the date of downgrade, but that is of no concern to us, since TALF purchases had to be held to maturity, and market-to-market pricing was not an issue.

The code below generates a dictionary of pandas df that you can pass to a solutions data frame to answer all your questions:

```
purch = pyabs.simulate_purchase_per_sim_rate_scenario(purchase_weeks=purchase_weeks,
```

```
purch = pjabs.simulate_purchase_per_sim_rate_scenario(purchase_weeks=purchase_weeks,
                                                      sims=sims,
                                                      rates_sim_dict=all_sims,
                                                      spreads_dict=spreads,
                                                      date_index=date_ix)
```

```
solution = pd.concat(purch)
```

```
solution.reset_index(inplace=True)
solution.drop(['level_1'], axis=1, inplace=True)
solution.rename({'level_0': 'simulation'}, axis=1, inplace=True)
```

```
solution.head(50)
```

| | simulation | asset | purchase_week | purchase_date | maturity_date | term | f_risk_capital | risk_capital | fed_loan | total_purchase | libor | spread_ove |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 1_yr_helc_AAA | 1 | 2009-02-20 | 2010-02-19 | 1 | 0.12 | 30 | 220 | 250 | 208 | |
| 1 | 0 | 3_yr_credit_card_AAA | 2 | 2009-02-27 | 2012-02-24 | 3 | 0.06 | 20 | 313 | 333 | 319 | |
| 2 | 0 | 2_yr_auto_AAA | 3 | 2009-03-06 | 2011-03-04 | 2 | 0.11 | 20 | 161 | 181 | 178 | |
| 3 | 0 | 1_yr_student_loan_AAA | 4 | 2009-03-13 | 2010-03-12 | 1 | 0.08 | 20 | 230 | 250 | 219 | |
| 4 | 0 | 2_yr_auto_AAA | 5 | 2009-03-20 | 2011-03-18 | 2 | 0.11 | 10 | 80 | 90 | 185 | |
| 5 | 1 | 3_yr_helc_AAA | 1 | 2009-02-20 | 2012-02-17 | 3 | 0.14 | 30 | 184 | 214 | 251 | |
| 6 | 1 | 1_yr_helc_AAA | 2 | 2009-02-27 | 2010-02-26 | 1 | 0.12 | 20 | 146 | 166 | 209 | |
| 7 | 1 | 1_yr_helc_AAA | 3 | 2009-03-06 | 2010-03-05 | 1 | 0.12 | 20 | 146 | 166 | 124 | |
| 8 | 1 | 2_yr_auto_AAA | 4 | 2009-03-13 | 2011-03-11 | 2 | 0.11 | 20 | 161 | 181 | 101 | |
| 9 | 1 | 2_yr_credit_card_AAA | 5 | 2009-03-20 | 2011-03-18 | 2 | 0.05 | 10 | 190 | 200 | 101 | |
| 10 | 2 | 3_yr_auto_AAA | 1 | 2009-02-20 | 2012-02-17 | 3 | 0.12 | 30 | 220 | 250 | 50 | |
| 11 | 2 | 1_yr_credit_card_AAA | 2 | 2009-02-27 | 2010-02-26 | 1 | 0.05 | 20 | 380 | 400 | 198 | |
| 12 | 2 | 2_yr_helc_AAA | 3 | 2009-03-06 | 2011-03-04 | 2 | 0.13 | 20 | 133 | 153 | 287 | |
| 13 | 2 | 1_yr_credit_card_AAA | 4 | 2009-03-13 | 2010-03-12 | 1 | 0.05 | 20 | 380 | 400 | 220 | |

```
def simulate_purchase_per_sim_rate_scenario(purchase_weeks, sims, rates_sim_dict, spreads_dict,
    """Simulate purchases per scenarios.

    This function simulates the purchase of assets under the different interest
    rates scenarios.
    """
    purchase_dict = {}
    # there is probably a better way to do this loop
    for i in tqdm(range(sims)):
        capital = []
        assets = np.random.choice(list(p_issuance.keys()), purchase_weeks, p=list(p_issuance.val
        terms = np.random.choice(list(p_term.keys()), purchase_weeks, p=list(p_term.values()))
        for asset, term in list(zip(assets, terms)):
            capital.append(f_risk_capital[asset][term])
        data_dict = {'rate_label': list(assets),
                     'term': list(terms),
                     'f_risk_capital': capital}
        purchase_dict[i] = pd.DataFrame(data_dict)
        purchase_dict[i]['purchase_week'] = purchase_dict[i].index+1
        purchase_dict[i]['purchase_date'] = date_index[purchase_dict[i]['purchase_week']]
```

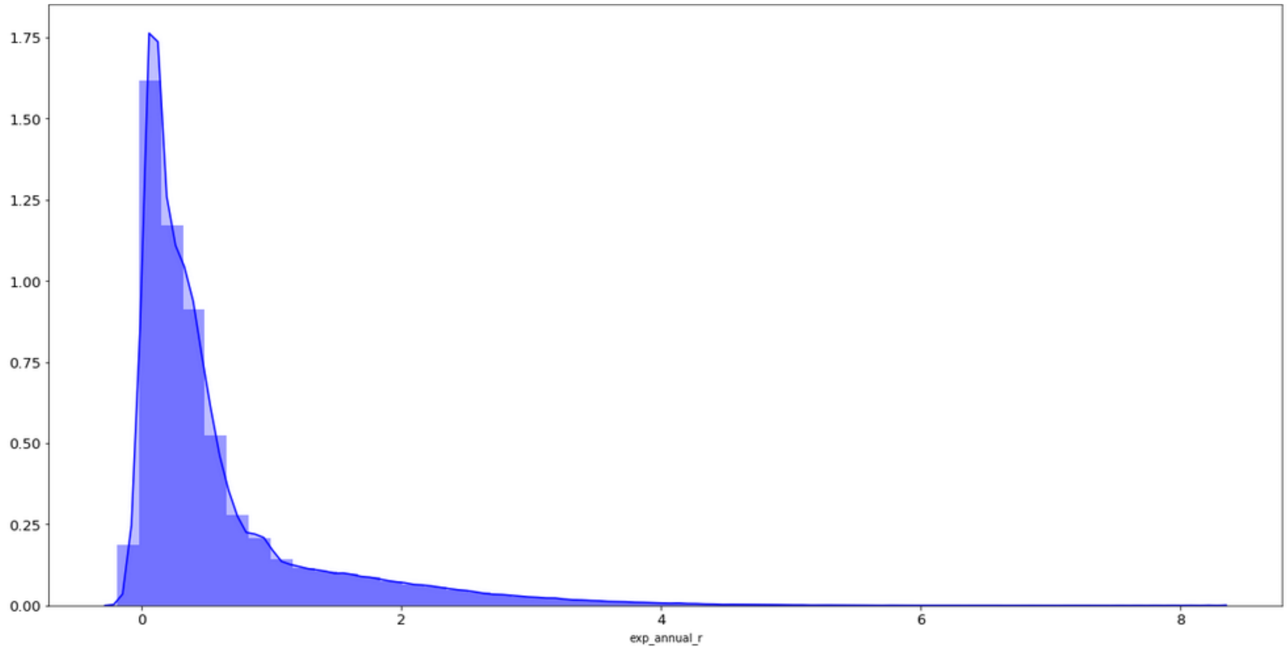
```python
        purchase_dict[i]['maturity_date'] = date_index[purchase_dict[i]['purchase_week']+(purcha
        purchase_dict[i]['asset'] = purchase_dict[i]['term'].astype(str) + '_yr_' + purchase_dic
        purchase_dict[i]['benchmark_asset'] = '3_yr_' + purchase_dict[i]['rate_label']

    # there is probably a way to merge this loop with the previous one
    for i in range(sims):
        for ix, col in purchase_dict[i].iterrows():
            purchase_dict[i].at[ix, 'benchmark_ABS_spread'] = rates_sim_dict[i].iloc[col['purcha
            purchase_dict[i].at[ix, 'libor'] = rates_sim_dict[i].iloc[col['purchase_week']]['lib
            purchase_dict[i].at[ix, 'risk_capital'] = to_invest[ix]
            purchase_dict[i].at[ix, 'fed_loan'] = (to_invest[ix]/col['f_risk_capital'])-to_inves
            purchase_dict[i].at[ix, 'final_rating'] = estimate_transition_vector('AAA', col['ter
        purchase_dict[i]['total_purchase'] = purchase_dict[i]['fed_loan'] + purchase_dict[i]['ri
        purchase_dict[i]['spread_over_libor'] = purchase_dict[i]['benchmark_ABS_spread'] - (purc
        r1 = ((purchase_dict[i]['libor'] + purchase_dict[i]['spread_over_libor']))*purchase_dict
        r2 = ((purchase_dict[i]['spread_over_libor'])-100)*(1-purchase_dict[i]['f_risk_capital']
        purchase_dict[i]['exp_annual_r'] = ((r1+r2)/(purchase_dict[i]['f_risk_capital']))/10000
        purchase_dict[i] = purchase_dict[i][['asset',
                                             'purchase_week',
                                             'purchase_date',
                                             'maturity_date',
                                             'term',
                                             'f_risk_capital',
                                             'risk_capital',
                                             'fed_loan',
                                             'total_purchase',
                                             'libor',
                                             'spread_over_libor',
                                             'exp_annual_r',
                                             'final_rating']]
        for col in ['risk_capital', 'fed_loan', 'total_purchase', 'libor', 'spread_over_libor']:
            purchase_dict[i][col] = purchase_dict[i][col].astype(int)

    return purchase_dict
```

ase_per simulated_rate_scenario.py hosted with ♡ by GitHub view raw

## Results

Distribution of returns and their percentiles:

- What is the distribution of expected annual returns if we commit USD $100MM to the purchase program?




- What is the probability of losing money if we commit to buying every bond we can with our capital, regardless of the spread at purchase time?

```
len(solution[solution['exp_annual_r']<0])/(sims*purchase_weeks)

0.049836
```

- How many Fallen Angels can we expect in TALF eligible securities? (transitions to BB, B, and CCC ratings)

```
solution['final_rating'].value_counts(normalize='True')

AAA     0.884480
AA      0.101102
A       0.011288
BBB     0.001516
BB      0.000852
B       0.000418
CCC     0.000214
D       0.000130
```

- What is the probability of making more than a 5% hurdle rate, if we discriminate which bonds to buy based on their spreads and leverage at the moment of purchase?

```
strategy_df = solution[solution['exp_annual_r']>0.05]
len(strategy_df)/(sims*purchase_weeks)

0.881436
```

- What is the median no loss annual return of that strategy?

```
strategy_df['exp_annual_r'].median()

0.37166140387288993
```

- What are the expected defaults in basis points?

```
(strategy_df[strategy_df['final_rating'] == 'D']
['total_purchase'].sum()/strategy_df.total_purchase.sum())*10000

1.2697864542783506
```

- What is the average fraction of capital we expect to commit for every $1 of purchases?

```
strategy_df['risk_capital'].sum()/strategy_df['total_purchase'].sum()

0.0854416650602756
```

- What is the average life of the portfolio?

```
sum(strategy_df['term']*strategy_df['total_purchase'])/sum(strategy_d
f['total_purchase'])

2.241573279710343
```

With all these results, the recommendation to the family office in February 2009 was to commit capital to the TALF program, expecting a 37% annual return for 2 to 3 years, in a no loss scenario. The basic strategy was only to invest if the bond in consideration under simulated market conditions with real pricing data (spreads, leverages, and maturities), allowed them to earn an annual return greater than 5%.

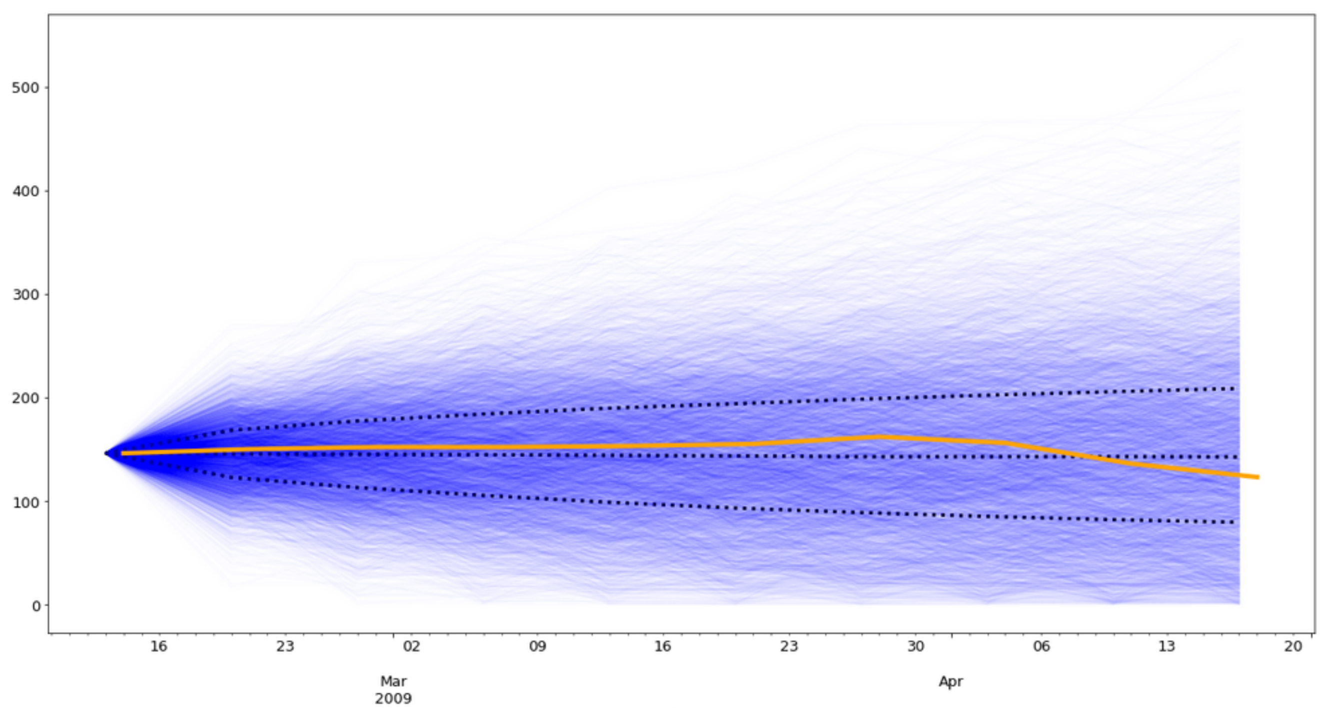
Expected defaults were modeled at less than 10 bps, and they could have expected to receive ~91.5 cents of leverage for every 8.5 cents committed to a diversified portfolio of TALF qualified ABS.

Interest rates and spreads stayed within the confidence intervals of the model.

## How well did my interest rate model fit reality? How well did the simulated correlations behave?

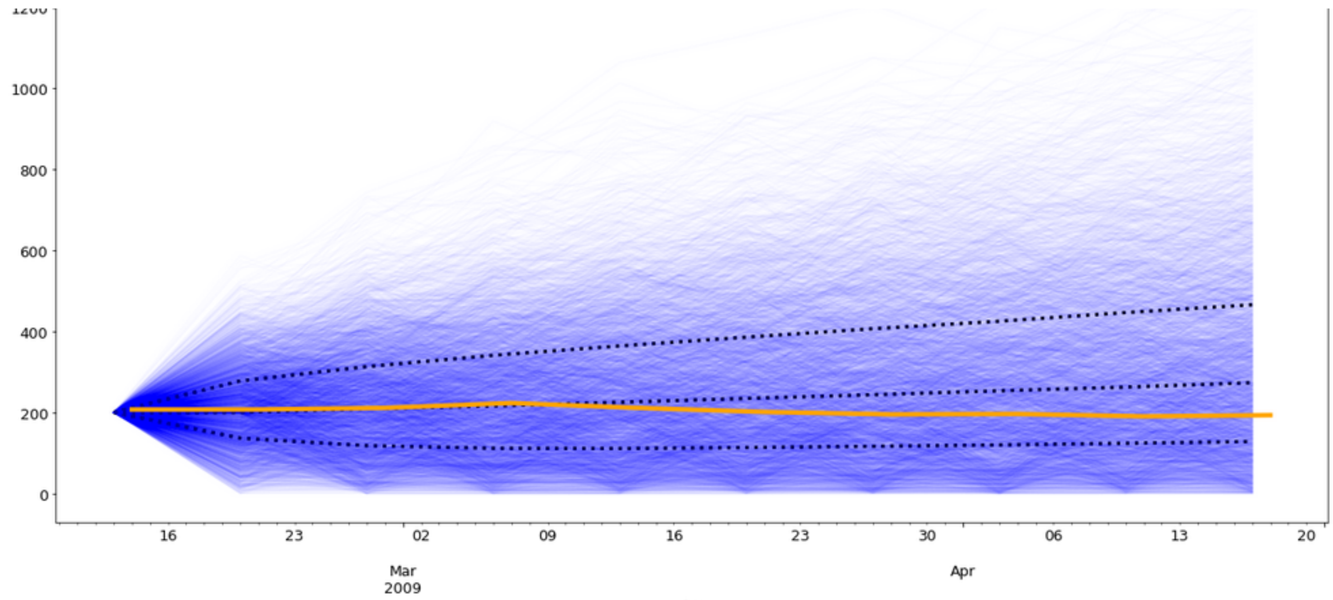
Below you can see the actual 3 year credit card AAA spreads in orange, plotted against many simulated paths, with dotted lines representing the median, 25th, and 75th percentiles.

Although the purchase strategy was concentrated in the first 4 weeks of TALF, the graphs shows 10 weeks of results. Realized rates stayed within expectations.



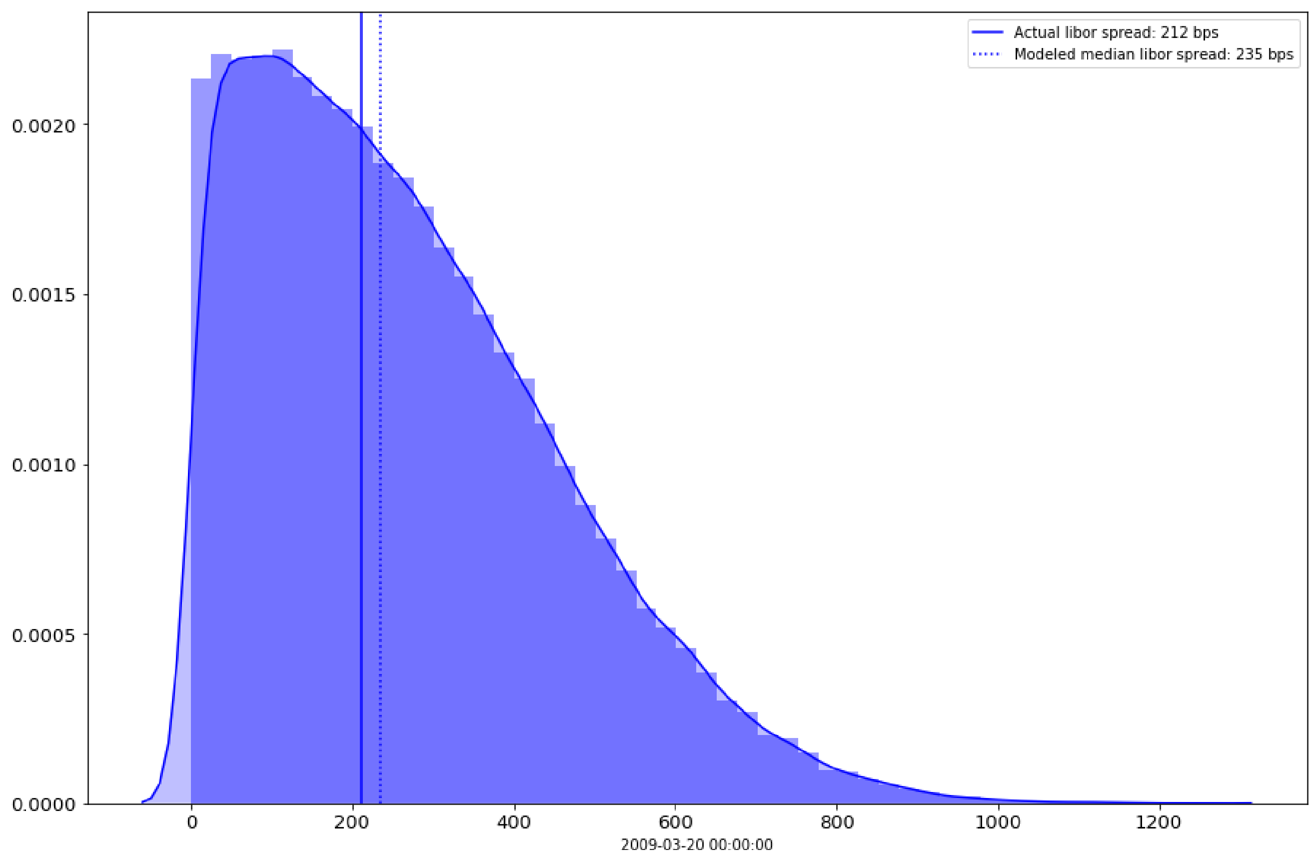

Simulated spreads for 3 years AAA student Loans

All other realized rates stayed close to the model's expectations.

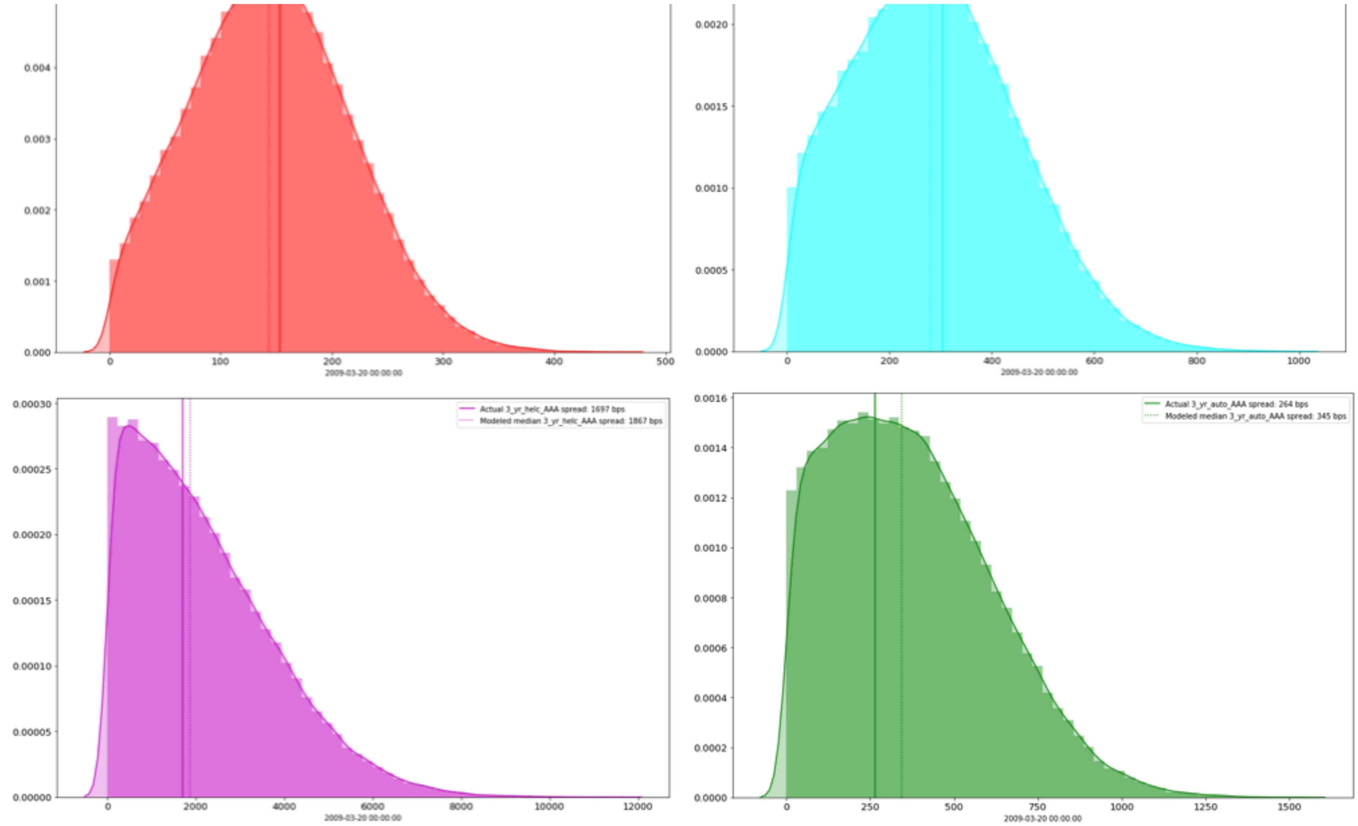



Simulated LIBOR rate

Another way to look at the graph above is to plot the distribution of spreads in the last week of our purchase program, and compare the actual values with the median values expected from our AR1 simulation model.

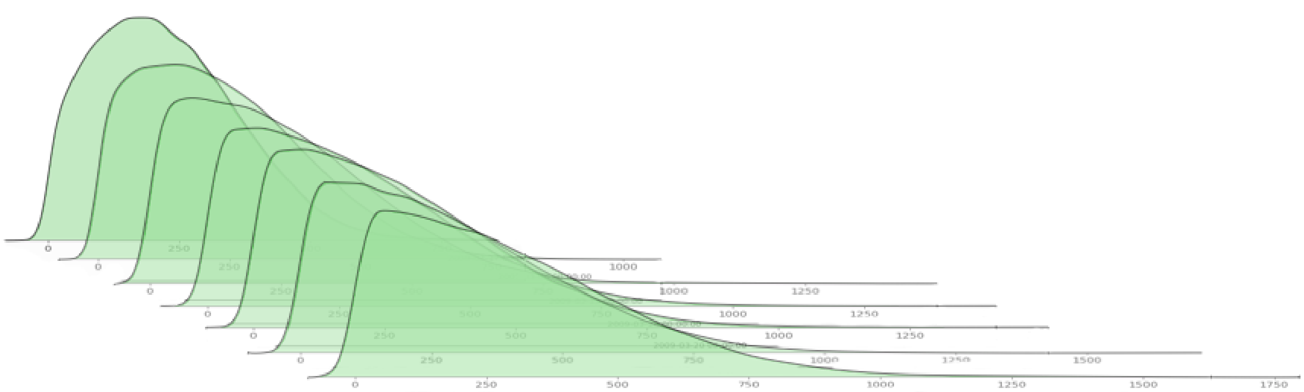



LIBOR



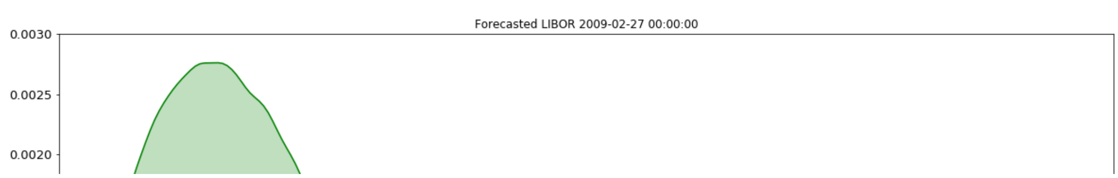

student loans, credit card, helc, auto

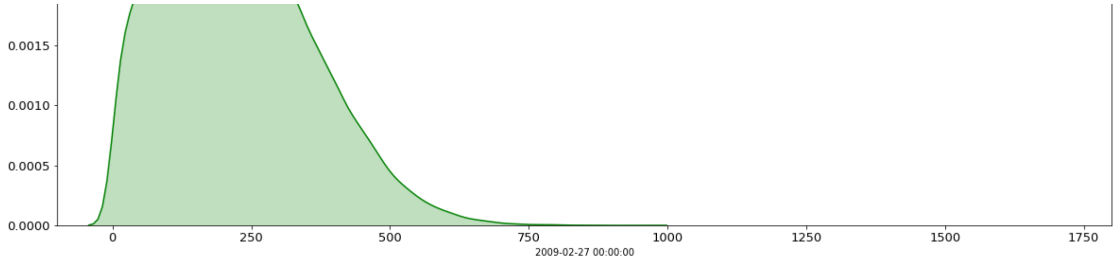
A better visualization, an animation of the shifting distributions of forecasted values for the LIBOR rate:



Time series: Distribution of forecasted LIBOR rates

As we move forward in time in our forecast, we start seeing longer tails in the shape of the distributions, but that did not affect our short term forecast.

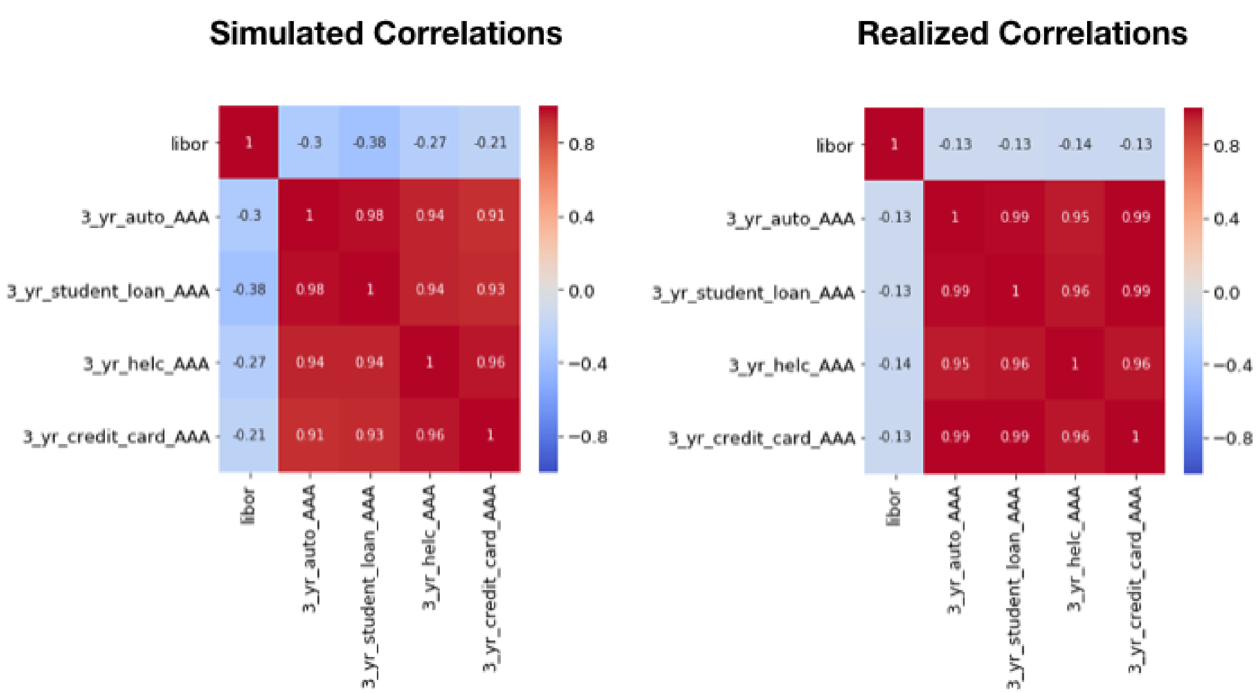





Animation showing distribution of expected LIBOR rates, from the Monte Carlo simulation and at different points in time

As we saw before, the median values of the simulation did a good job estimating the actual realized values.

Simulated correlations also stayed within expectations. Below is a chart showing correlations for one of the 100,000 simulated scenarios chosen randomly, vs realized correlations.

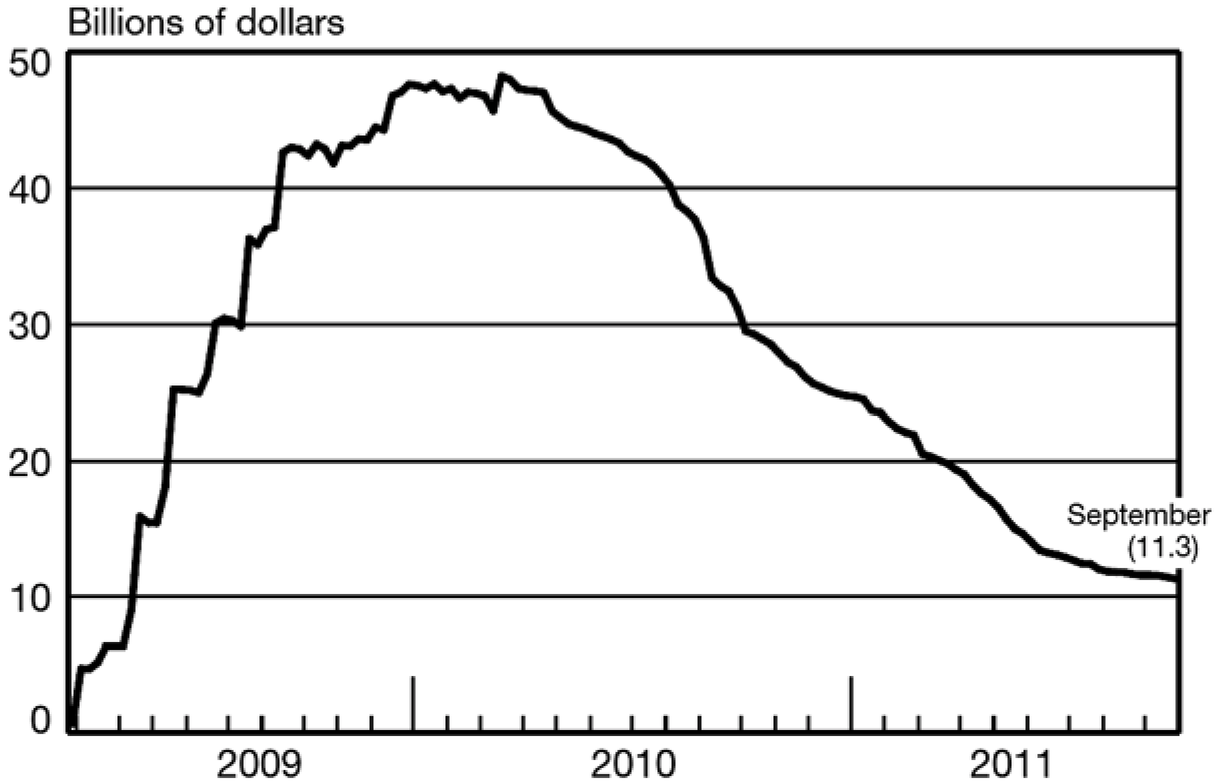



Simulated vs realized correlations

Defaults and transitions stayed within modeled expectations.

Loans were available for about 3 years; however, the juiciest returns were made investing in the first year of the program, when spreads were high relative to later years.

To my knowledge, no investor experienced losses.




Outstanding TALF loans from inception of program to end. Source: Federal Reserve Statistical Release

## In retrospect, was the TALF program good?

A paper published a while ago by the Federal Reserve Bank of Chicago concluded that the TALF program prevented the US economy from sinking deeper than it did.

The paper references a study done by some academics where they found a strong link between financing conditions and the sale of vehicles when using both household level data and aggregate data.

Specifically, they found that 38% of the decline in vehicle sales between 2007 and 2009 could be attributed to increases in the interest rates on new vehicle loans and households' perception that credit conditions were unfavorable.

The purchases of households that were likely to face borrowing constraints were extremely sensitive to changes in credit conditions, but were not sensitive to expected changes in income. The study found that aggregate vehicle sales fell 130,000 units for every 1 standard deviation increase to the interest rate.

The study suggested that (directly or indirectly) by making credit more accessible and affordable to consumers, TALF supported vehicle sales and the economy as a whole.

## Are there other TALF-like opportunities out there? How can we find them?

There are always opportunities associated with a financial crisis, but as far as I am concerned, domain expertise, fast prototyping/coding, and fast decision making are the only ways to spot and take advantage of those opportunities in finance.

There is no all-purpose AI (yet) that will point out investments in special opportunities around credit risk, commodities, activism, etc. All those "AIs for Credit Risk" out there developed by startups are just repurposed Scikit-Learn and/or Google's TensorFlow code mostly stitched together by machine learning engineers with very little or zero domain expertise in finance. Their claims look good to a non-quant audience, but I guarantee you: their claims will not hold in a stressed, real-life scenario. These developers do not analyze tail risk and shifting correlations: they don't know how.

In many cases, the developers do not even know that some of their creations have built in technical bugs. For example, in his book "Advances in Financial Machine Learning," Dr. Marcos López de Prado documents a bug in Scikit-Learn's cross-validation. That piece of code is built-in in many of the AI's for credit risk I have seen out there.

You can check out the issue below:

**Scoring functions don't know classes_ · Issue #6231 · scikit-learn/scikit-learn**

Moving the discussion with @amueller from pydata/patsy#77 (comment). Proposing to: add an optional 'labels' argument to...

github.com

But of course, there are some exceptions and a few products out there are good.

About unique investment opportunities, yes, there are a few interesting ones I am planning to write about in future articles and after the opportunity has been arbitraged out. For obvious reasons, it is unwise to write about specific opportunities while they are live.

Research – Fixed Income

SAGA Capital

## US Government Creates Opportunities for Investors in Fixed Income Instruments

A detailed analysis performed by SAGA Capital (SAGA) has identified investing in certain types of Asset Backed Securities (ABS) as an attractive, low-risk way to achieve above-average returns provided the right strategies are in place from the start.

For this, the management of the enterprise has to be in the hands experienced bankers with backgrounds in structured finance, coupled with up to 20-1 leverage provided by the US Federal Government via the Term Asset Backed Loan Facility (TALF). This could translate into 20-40% IRRs via investment vehicles that SAGA is setting up.

Market conditions are very good for investing in some sectors of the US ABS market, despite the popular perception of it being a risky business. Aside from the sub-prime mortgage market and some other related asset classes, default rates have remained within expected values.

Inherent financial & regulatory constraints in the bureaucratic and tradition-bound structures of European and US investment banks are not allowing the banks to actively participate in this opportunity. The opportunity we will present here is better suited for high net worth individuals and a few institutional investors not bound by government constraints and who can move fast to seize the opportunity while it is still present.

Before explaining how to specifically profit from this opportunity, a basic understanding of ABS, TALF procedures, and specific macroeconomic conditions of the world and the US in particular is necessary.

### What is a Securitization?

Securitization is a widely accepted form of structured debt that is repaid solely by cash flows from a pool of isolated assets

While the securitization market has traditionally embraced consumer and commercial assets such as auto loans, credit card receivables, mortgages and equipment leases, the market has also accommodated esoteric risks such as project finance assets, natural catastrophes, man-made catastrophes (terrorist attacks), life settlements, and royalty and intellectual property assets.

In a basic securitization, assets are sold by an entity ("Originator") to a bankruptcy-remote special-purpose vehicle ("SPV"), usually set up in an offshore jurisdiction. The SPV then issues bonds into the capital markets, whose payment of principal and interest are tied to the performance of the assets. The proceeds received through the bond issuance are passed to the Originator via the SPV (See figure 1 for an example of an hypothetical USD 1bn securitization).

Once the bonds have been retired, the

*This report is intended only for the personal and confidential use of the designated recipient(s), and to be discussed outside the territory of the United States of America. If you are not the intended recipient of this message you are hereby notified that any review, dissemination, distribution or copying of this message is strictly prohibited. This communication is for information purposes only and should not be regarded as an offer to sell or as a solicitation of an offer to buy any financial product, an official confirmation of any transaction, or as an official statement of SAGA Capital, LLC. We do not represent that this information is complete or accurate and it should not be relied upon as such. All information is subject to change without notice.*

February, 2009

Page 3 of 6 Research – Fixed Income



Figure 2: US Household Debt as a % of PDI

administration.

The importance of US consumer spending and its relationship to US GDP is little understood in the context of the world economy (See figure 3). The US economy is by far the largest economy in the world, and US consumer spending represents a staggering 70% of the US GDP.

In the last 15 years, the easy credit markets experienced in the US were fueling most of the GDP growth experienced in the US and the rest of the world, in a sort of cycle, as shown in figure 4. This worldwide industrial growth, fueled by US consumer spending, had a direct impact on the corporate profits of the world markets, as evidenced by the growth of market capitalization of virtually every stock exchange in existence.

In mid 2007, one type of asset class that had been growing dramatically within the securitization market started to show signs of problems. The asset class was the sub-prime mortgage, which had become very popular among investment banks, pension funds, and hedge funds, due to its supposed low risk and attractive returns. The reason why the estimation of risk of this asset class was out of line with reality is beyond the scope of this report, however, its dramatic failure collapsed the appetite not only for this asset class, but for all other asset classes that had been securitized.

As figure 5 shows, the spread for a typical 5-year, "AAA" rated securitization increased from about 0.2% over LIBOR in early 2007, to 6% over LIBOR in December 2009. Similarly, the volume of securitizations in the market decreased from over USD 1.2 trillion in 2006, to less than USD 30 billion in the first quarter of 2009, a 98% **drop with respect to its** peak (see figure 6). This drastic decrease in the available funds for financing, and its associated increase in the cost of financing created a global financial "gridlock" that needed to be handled by world authorities.

Early this year, the IMF estimated a 1.3% drop in global output in 2009, and a sluggish recovery in 2010, and stated that "by any measure, this downturn represents by far the deepest global recession since the great depression." These events meant that credit to the private sector of the world economy would for sure decline, unless



Figure 3: Top GPD's (2007), in trillions of USD

The TALF opportunity analyzed by my advisory firm back in early 2009

As a matter of fact, practically all the analysis and modeling you learnt about here was done over 10 years ago, when the TALF was announced. I was actively trying to raise capital to take advantage of the opportunity, and the background information plus all the simulations in this article were part of my presentation to interested parties.

Currently, I am writing proprietary code for my private repos in Bitbucket and analyzing several opportunities that are as exciting as the TALF was back them.

But aside from potential trading and structuring events, some of the best opportunities to profit from data science I have seen are in the raw data stored in silos of large corporations, and government agencies, which can be used to generate non trivial sources of new revenue. In most cases, the corporations that own these gold mines of data are doing nothing about it for many reasons.

I hope this article helped you understand one of the best investment opportunities derived from policies during the Obama administration, and how simulation models can help decision making under uncertainty.

You can find all the code presented in this article in my Github repo, as well as a Jupyter notebook here with examples of use.

If you enjoyed this post, please let me know, share this story, and leave your feedback below.

```
All the images, code, and graphs in this article belong to and/or
have permission for distribution.
```

. . .

Sources:


1. The US Government Creates Opportunities for Investors in Fixed Income Instruments, Fixed Income Research — Special Report. SAGA Capital, February, 2009.

2. The asset-backed securities markets, the crisis, and TALF. Sumit Agarwal, Jacqueline Barrett, Crystal Cun, and Mariacristina De Nardi. Federal Reserve Bank of Chicago.

3. Term Asset-Backed Securities Loan Facility: Frequently Asked Questions. Federal Reserve Bank of New York, 2009.

4. Federal Reserve Statistical Release H.4.1, "Factors Affecting Reserve Balances," Table 1.


Finance Data Science Bonds Machine Learning Trading