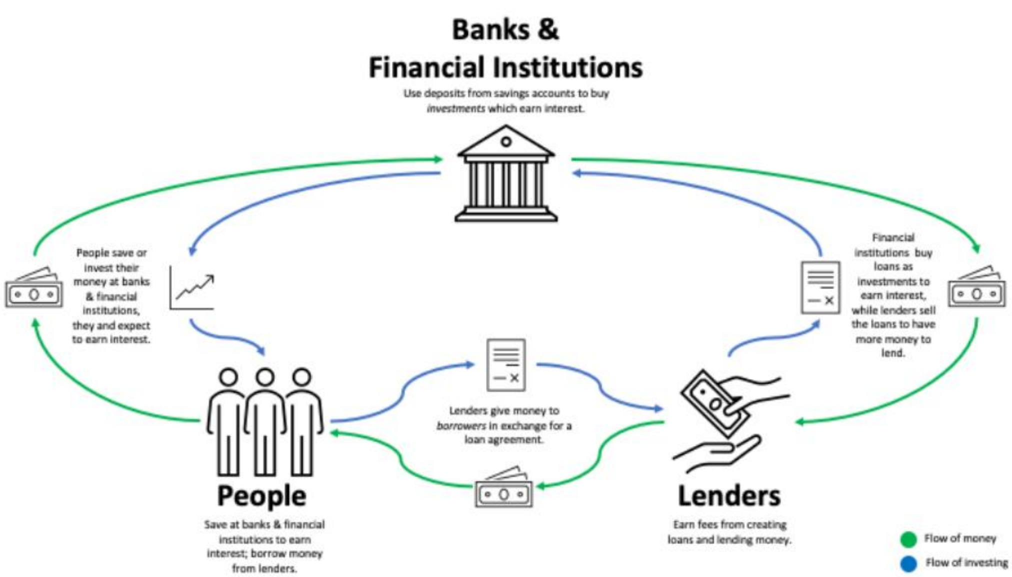
Power BI: Mortgage Trading Analysis

Project & Data Introduction:

- In this project, I execute a trade of mortgages in the Capital Markets
- The datasets used for this analysis are based on real mortgage and market data
    - Derived from de-identifying over 5000 mortgages
- Financial System is a collective system of people, banks, financial institutions and lenders working together to move money around in the economy
    - People who have extra money will invest in banks and financial institutions (Savers)
    - People who need extra money get it from the banks (Borrowers)
- Loan Agreement is a promise to repay the loan amount (usually with interest)
- Lenders are happy to write loan agreements since they make money on the fees, they charge the borrowers
    - When lenders write loan agreements, they sell the loans in Capital Markets to Banks and Financial Institutions. When they sell the loan, they have the money again to write more loans and continue earning fees
    - Some banks also act as lenders
- Banks and Financial Institutions are happy to purchase loan agreements because they see the loan as an investment that earns interest overtime with each monthly payment the borrower makes
- For example, a 30 Year loan for $100K at 5% annual rate pay $195K total over the life of the loan ($95K in interest)
- Some interest earned in investment is returned to the savers

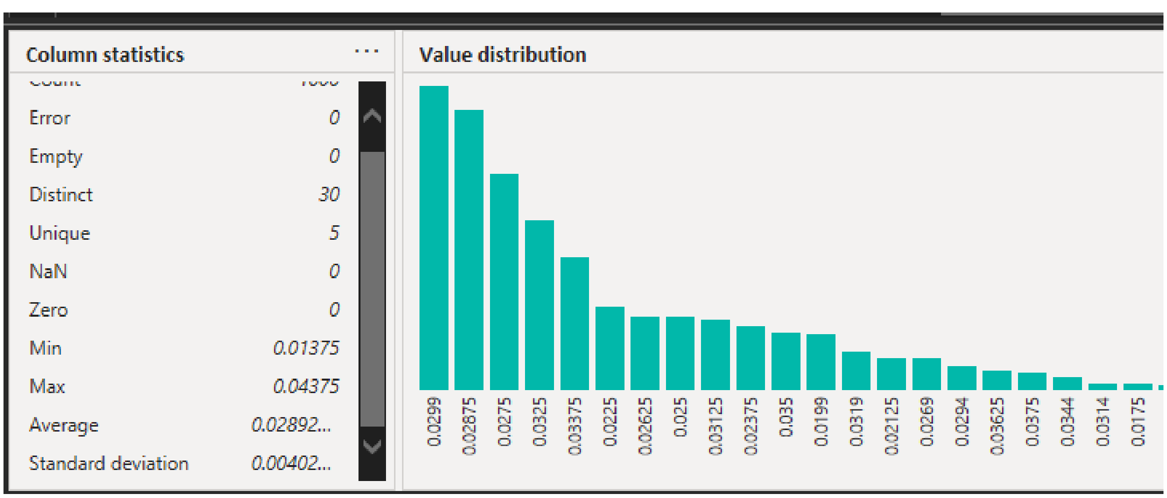


- A borrower qualifies for the mortgage based on their Credit Score, Debt to Income ratio (DTI) and Loan to Value (LTV)
- Ideally, the leader would like to sell the loan to Banks as soon as possible to get the money back and keep lending. Before the loan can be traded, it must be audited for errors or missing documentation. Once audited, it is sent to the document custodian to keep until the ownership is transferred

- Mortgage debt in the US makes up over $12 Trillion as of 2023, which is 75% of all household debt
- When mortgages are sold to investors, they are often combined into groups called "Pools". A pool can be one or more mortgage loans
- Mortgage trading can be of two forms –
  - **Whole Loan Trade** – When the investor/bank/institution picks and bids on the mortgages/loans individually. This type of trade is inefficient as the bank must analyze each loan before bidding
  - **Securitization** – Mortgage pools are bundled into Mortgage Back Securities (MBS). The investor/bank/institution buys the entire security instead of the individual loans. Here the investor doesn't have to analyze each loan in the MBS, but just the performance of the MBS bond. Securitizations are usually formed by combining mortgages with similar characteristics (like rate, term, loan-to-value, lender, etc.). Not every loan is eligible for securitization as the loan must meet the requirements set forth by the Securitizer. Fannie Mae and Freddie Mac are two of the largest US securitizers. They purchase 70% of all mortgages in the US and pool them into securities which they call UMBS (Uniform-MBS)
- Once an investor has identified the bond/MBS they are willing to purchase, they place a bid
  - Trade Price is expressed as a % of the bond's principal balance
  - Trade Amount = Trade Price * Principal Balance
  - Trade Premium = Trade Amount – Principal Balance
  - For example, if the Trade Price is 101% then the investor is willing to pay $101K for an MBS with a principal balance of $100K. This gives a Trade Premium of $1K

<u>Analysis:</u>

1. Cleaned loan dataset in PowerQuery by creating calculated columns, replacing errored cell values and changing datatypes
2. Used **Column Profile** functionality in PowerQuery to analyze the distribution of values in various fields

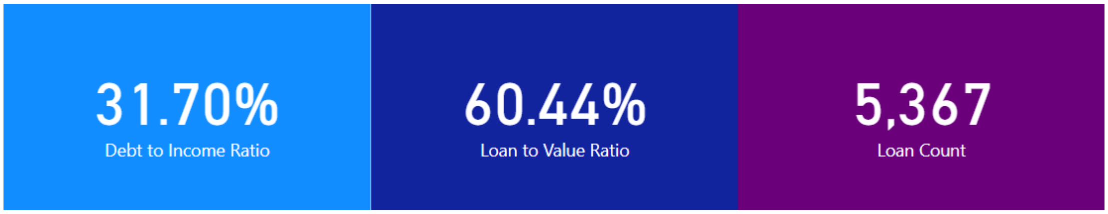


3. Created DAX Measures to estimate the Loan-to-Value and Debt-to-Income ratio metrics to assess the riskiness of the loans. Since some measures required multiple steps, I created multiple DAX variables to store interim results before calculating the final metric

```
Loan to Value Ratio = DIVIDE(SUM(loan_data[loan_amount]), SUM(loan_data[property_value]))
```

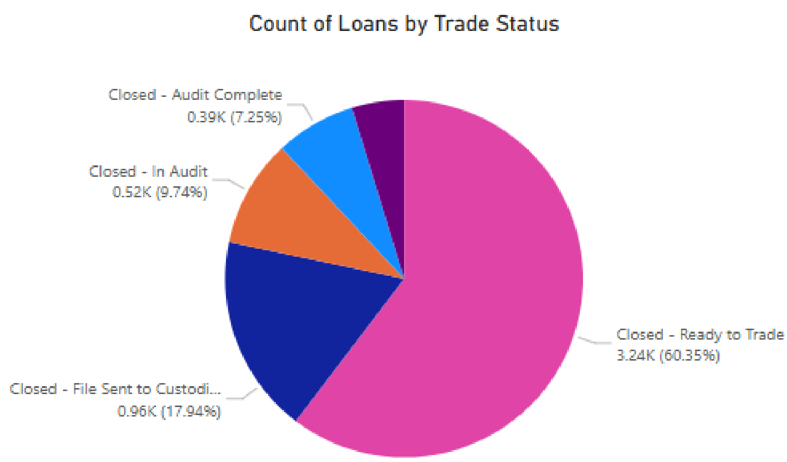
```
Debt to Income Ratio =
VAR monthly_income_in_thousands = SUM(loan_data[income_thousands])/12
VAR monthly_income = monthly_income_in_thousands * 1000
RETURN DIVIDE(SUM(loan_data[recurring_monthly_debt]), monthly_income)
```

4. The weighted DTI and LTV metrics across the 5,300+ loan seemed to be low. This tells me that the overall loan population has a low default risk. Usually, the LTV should be < 100% and DTI should be <50%.



5. Used **SWITCH DAX function** to create the status of each loan based on the last reported step completion date. Based on the status, I found that over 3,200 loans (60.35%) of loans have completed the audit, are with the custodian and ready to be traded.

```
Trade Status = SWITCH(
    TRUE()
    , ISBLANK(loan_status[file_in_audit]), "Closed - Needs Audit"
    , ISBLANK(loan_status[file_audit_complete]), "Closed - In Audit"
    , ISBLANK(loan_status[file_sent_to_custodian]), "Closed - Audit Complete"
    , ISBLANK(loan_status[file_at_custodian]), "Closed - File Sent to Custodian"
    , "Closed - Ready to Trade"
)
```



6. Since we are looking to sell some of these loans in 30 days and some borrowers will be making their monthly payments before the loans are traded, I estimated the outstanding principal balance after the monthly payment using the **PPMT DAX** function that calculates the amortized principal payment for a given payment period.
    a. Since the interest rate in the dataset is annualized, to convert it to the monthly interest rate I divided it by 12
    b. Payment Period tells me that the borrower has previously made N payments on the loan amount (N = 0, 1, 2, ..., Loan Term)
    c. Loan Term is the total number of terms (months) the loan is expected to be paid (usually 120, 180, 360 months)

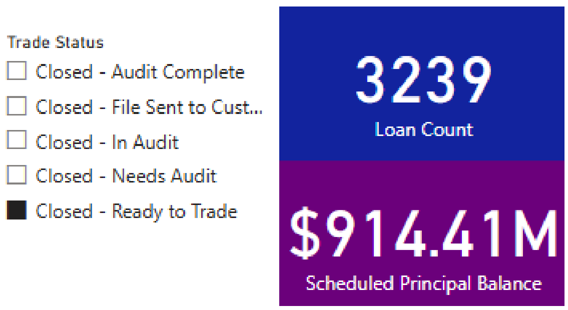
d. Current Balance gives me the remaining balance on the loan amount. For example, if a borrower has a $100K loan and has made 1 previous payment of $1000 of which $500 is principal and $500 is interest, then the current balance is $99,500

```
Amortization Amount = PPMT(loan_balance[interest_rate]/12, loan_balance[Payment Period], loan_balance[loan_term], loan_balance[current_balance], 0, 0)
```

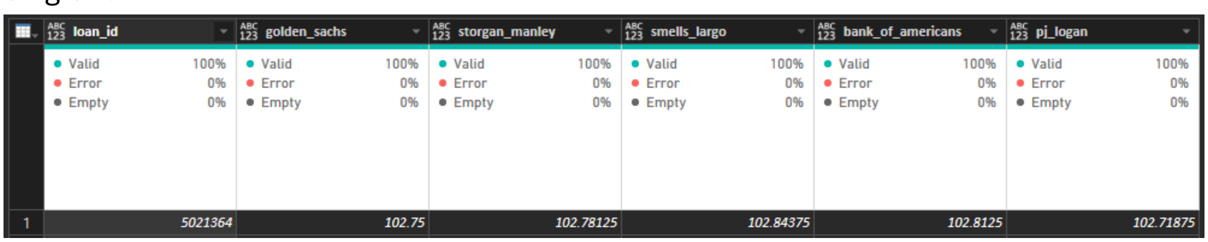
7. I noticed that for some loans the next payment date is past the 30-day period within which we expect to sell the loan. So, I updated the code to only subtract the amortization amount if the next payment date on the loan is before our expected trade date.

```
Scheduled Principal Balance =
    loan_balance[current_balance] +
    IF(loan_balance[next_payment_due_date] < loan_balance[Scheduled Next Payment Date]
    ,  loan_balance[Amortization Amount]
    , 0)
```

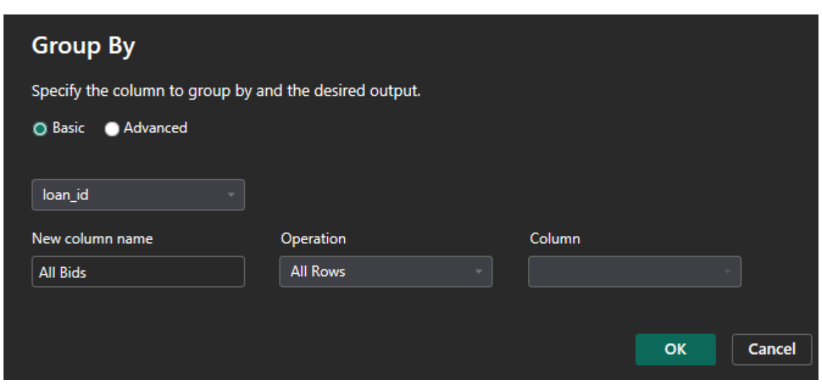
8. Since the Lender is only focused on trading loans that are post audit and ready to trade, I created some visuals to display the total count and $s associated with loans ready to trade.



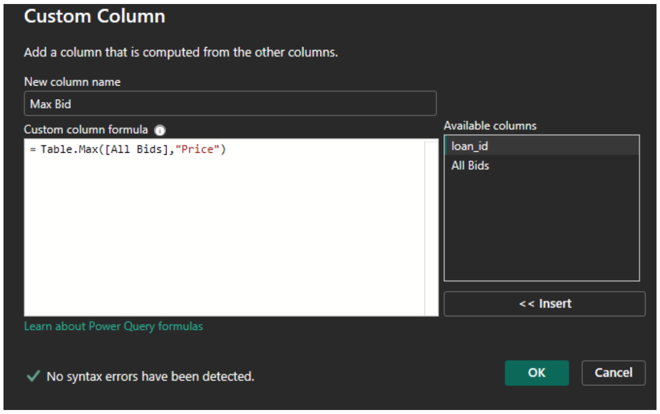
9. Transformed on Investor Bids dataset in **PowerQuery** –
   a. The dataset was structured in a way that bids from different investors for a loan were in a single row



   b. First, I **unpivoted** the dataset so that we have three columns – Loan Id, Counterparty and Price
   c. Then I used **Group By functionality in PowerQuery** to aggregate all the bids for a particular loan id into a single table using the **All-Rows operation**. This aggregates all the bids for a loan into a table



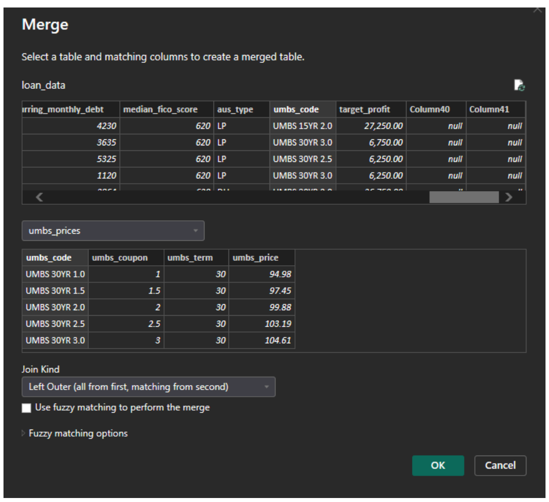
d. To find the highest bidder for each loan, I wrote **a Custom Column** in PowerQuery to find the max row in each table using the **Table.Max function**



e. The resulting dataset gives the highest bidder's name and price for each loan

10. To confirm if the bids we received as part of the Whole Loan trades are good, I compared the rate the lender would have received if the lender chose to go on the Securitization route (UMBS).

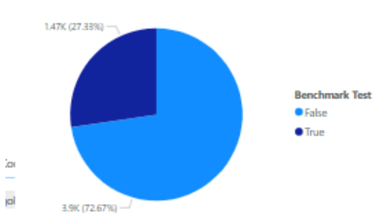
a. In PowerQuery, I **merge**d the loan dataset with the UMBS dataset that gives us the UMBS price based on term and UMBS coupon



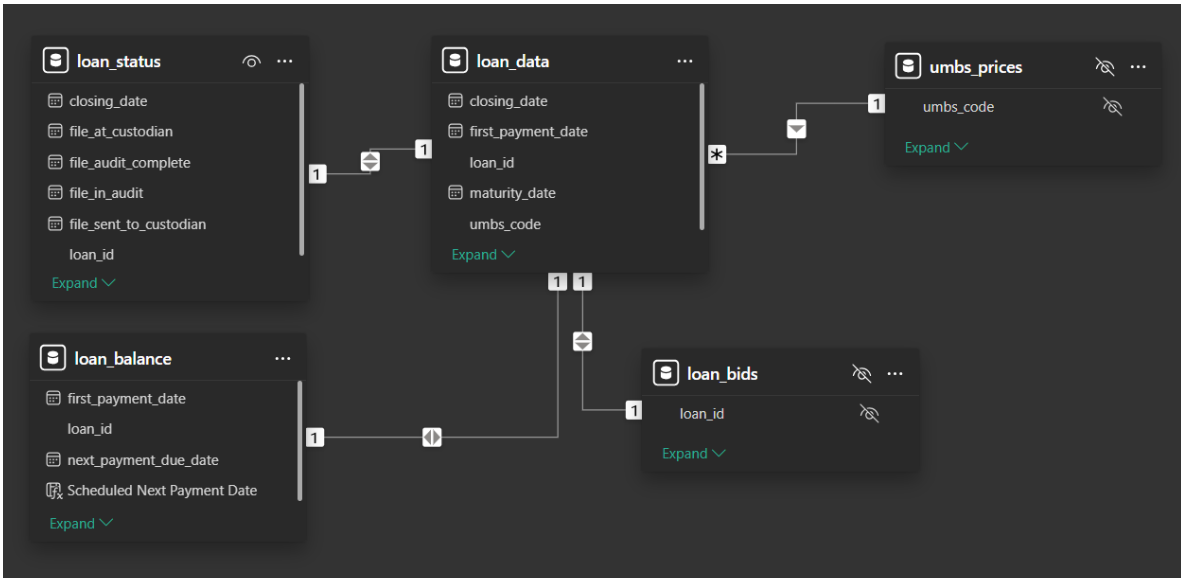
b. Next, I joined the cleaned Loan Bids dataset from the previous step to get the best Whole Loan price for each loan (merged on the loan_id field)

11. To maximize profit for the lender, I assumed that the lender would choose to sell the loan to Whole Loan Trade if the bid price is higher than the UMBS price. Based on the dataset I created in the previous step by joining the individual bids with the UMBS price, I wrote a DAX calculated column to flag how often the individual trade out-bidded the UMBS price. Approximately, 27.33% of loans have higher bids from the Whole Loan trade compared to the UMBS trade

```
Benchmark Test = IF(
    loan_data[Price] > loan_data[umbs_price]
    , TRUE()
    , FALSE())
```



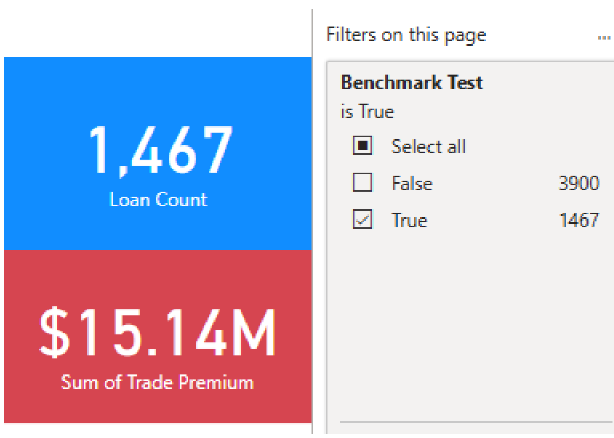
12. To calculate the profit/loss for the trades, I need the remaining balance on each loan. That field exists at a different table. I created a **data model** linking up all the different datasets using the identifier fields. Most of the tables are linked through **one-to-one bidirectional** relationships.



13. Given that all the tables are related in the backend, I used the **RELATED DAX** function to source the Principal Balance of each loan from the loan_balance dataset into the loan_data dataset. I multiplied the remaining balance by the Whole Loan Trade price to get the price the investor is willing to buy the loan

```
Trade Price = RELATED(loan_balance[Scheduled Principal Balance]) * (loan_data[Price]/100)
```

14. Then I calculated the Trade Premiums for all loans where the Whole Loan price is higher than the UMBS rate. If we sell the 27.33% of loans where the individual Whole Loan trade price out-bidded the UMBS price, we will make an additional $15.1M in revenue



15. I analyzed the average price I received across all bids in the whole loan trade. While a straight average gives us a directional answer, the weighted average would be more accurate since it weighs loans with higher balance more than loans with lower balance. To do this, I used the SUMX DAX function and weighed the loan price by the remaining open balance on each loan

```
Weighted Avg Price =
var numerator = SUMX(loan_data, loan_data[Price] * RELATED(loan_balance[Scheduled Principal Balance]))
var denominator = SUM(loan_balance[Scheduled Principal Balance])
RETURN DIVIDE(numerator, denominator)
```

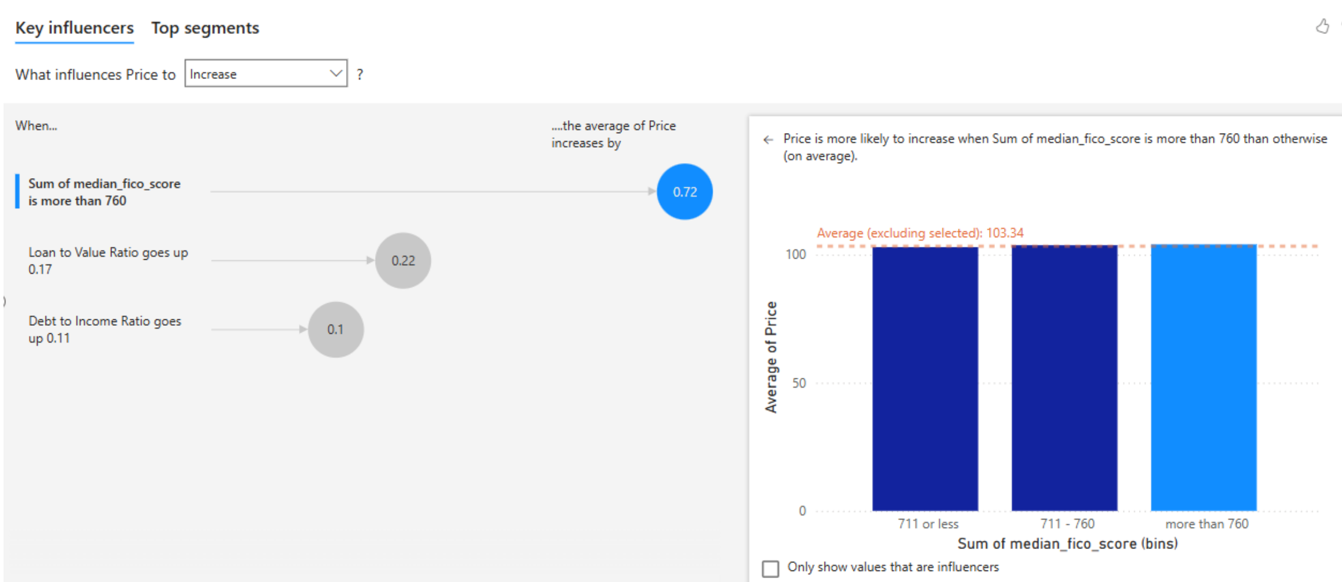
Since the weighted average is lower than the regular average, this tells us that loans with higher balance have slightly lower Prices on average compared to loans with lower balances.

16. To estimate the total profit on the loans, I added the origination charges to the trade premium and subtracted any lender credits we might offer to the borrowers. After that I divide the remaining value by the open balance on the sold loans to get the Profit Margin %

```
Loan Profit Margin = DIVIDE([Loan Gross Profit],SUM(loan_balance[Scheduled Principal Balance]),0)
```

Though the lender has a 3.66% weighted margin after selling the mortgages to the highest bidder, after adding the origination charges and lender credits, the margin jumps to 7.073% on all loans that are ready to trade

17. The goal of the lender was to achieve a 5% margin, since we made more than that in the current bunch of trades, I was asked to explore why the margin was higher than expected. Given that mortgage lending is a very competitive business, even a 0.25% difference in rates could lose us a potential borrower. I used PowerBI's built in Key Influencers visual to find patterns in loans that had higher price from a Bank/Institution. I found that the biggest influencers on price if the FICO score, where loans with a FICO score > 760 had 0.72% higher bid price than average and loans with FICO score below 711 had 0.85% lower bid prices than average.



Definitions

- **Loan-to-Value (LTV):** This gives the ratio of the loan amount to the property value. This metric tells the investors/lenders how much they can expect to get back if the borrower defaults on the loan. When the borrower defaults, the investor/lender gets to take control of the property, sell it and recoup their loan amount

- **Debt-to-Income (DTI):** This gives the monthly debt the borrower owes, divided by their monthly income. This tells the investors/lenders how likely the borrower will pay back the loan. Usually, investors/lenders expect to see this metric below 50%
- **Loan Profit Margin:** Profit divided by Loan Amount