Lab: Introduction to Watson OpenScale

Introduction

Businesses today are increasingly certain that AI will be a driving force in the evolution of their industries over the next few years. Yet for every successful AI project, there are many that fail to reach widespread adoption in the business and achieve their expected outcomes. This is partly because the mechanics of AI deployment can be complex, and there are still gaps in skills and tooling that can make it difficult for data science, IT operations, and business teams to work in lockstep. But beyond the operational challenges, there are also much more profound issues of trust and transparency that businesses need to address before they can truly turn AI into a business advantage.

Knowledge workers must be able to trust AI and explain the decisions it helps make before they will incorporate it in their business processes. If AI is a black box that simply takes in data and produces obscure, unexplainable outcomes, then there is no way for the business to judge whether these systems are producing fair, accurate outcomes, or have confidence in AI's ability to augment decision-making. Equally, the business will not be able to explain outcomes to customers, auditors, or compliance teams.

IBM Watson OpenScale is an open platform that helps remove barriers to enterprise-scale AI. Watson OpenScale enables the enterprise to:

- Measure performance of production AI and its impact on business goals
- Track actionable metrics in a single console
- Explain AI outcomes
- Detect and mitigate harmful bias to improve outcomes
- Track model drift
- Accept feedback to compute accuracy measures
- Accelerate the integration of AI into existing business applications.

Objectives

The goal of this lab is to familiarize the user with the features of Watson OpenScale. After completing this lab, you will understand how to:

1. Access Watson OpenScale
2. View Fairness and Quality Metrics
3. View Drift Metrics.
4. Explain a Transaction.
5. Compare Pre-production Models and Generate a Report.

Lab Use Case

Traditional lenders are under pressure to expand their digital portfolio of financial services to a larger and more diverse audience, which requires a new approach to credit risk modeling. Their data science teams currently rely on standard modeling techniques - like decision trees and logistic regression - which work well for moderate datasets and make recommendations that can be easily explained. This satisfies regulatory requirements that credit lending decisions must be transparent and explainable.

To provide credit access to a wider and riskier population, applicant credit histories must expand beyond traditional credit, like mortgages and car loans, to alternate credit sources like utility and mobile phone plan payment histories, plus education and job titles. These new data sources offer promise, but also introduce risk by increasing the likelihood of unexpected correlations which introduce bias based on an applicant's age, gender, or other personal traits.

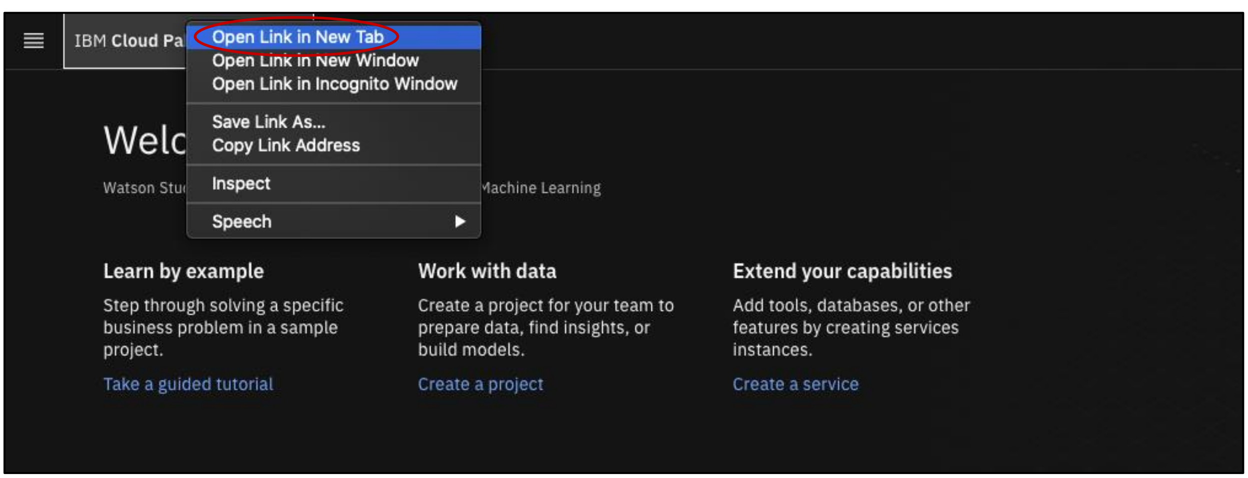
The data science techniques most suited to these diverse datasets, such as gradient boosted trees and neural networks, can generate highly accurate risk models, but at a cost. Such "black box" models generate opaque predictions that must somehow become transparent, to ensure regulatory approval such as Article 22 of the General Data Protection Regulation (GDPR), or the federal Fair Credit Reporting Act (FCRA) managed by the Consumer Financial Protection Bureau.

The credit risk model provided in this tutorial uses a training dataset that contains 20 attributes about each loan applicant. Two of those attributes - age and sex - can be tested for bias. For this tutorial, the focus will be on bias against sex and age.

Watson OpenScale will monitor the deployed model's propensity for a favorable outcome ("No Risk") for one group (the Reference Group) over another (the Monitored Group). In this tutorial, the Monitored Group for sex is `female`, while the Monitored Group for age is `19 to 25`.

Access Watson OpenScale.

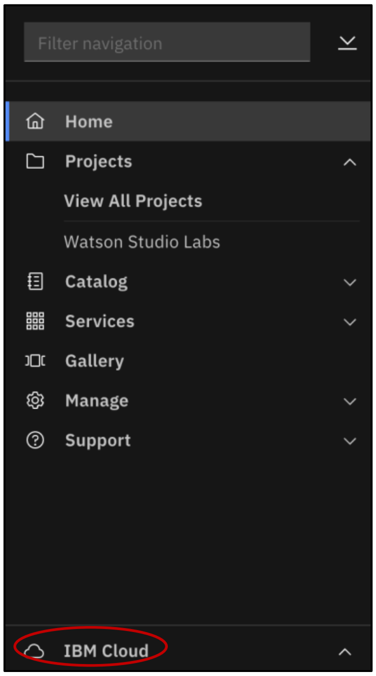
1. Right-click on the **IBM Cloud Pak for Data** label and click on **Open link in New Tab**.



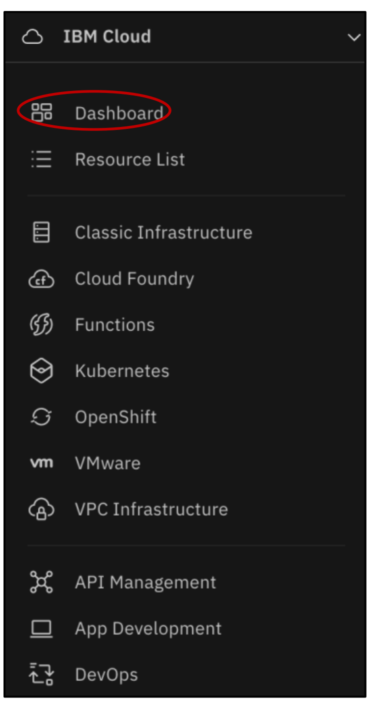
2. Click on the new **IBM Cloud Pak for Data** browser tab.

3. Click on the hamburger icon in the top left corner.

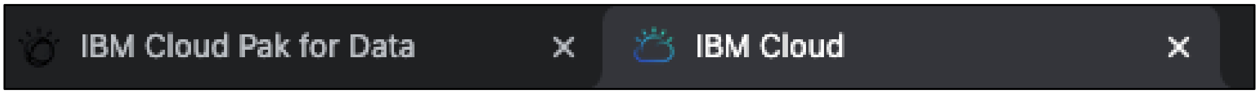
4. Click on **IBM Cloud**.



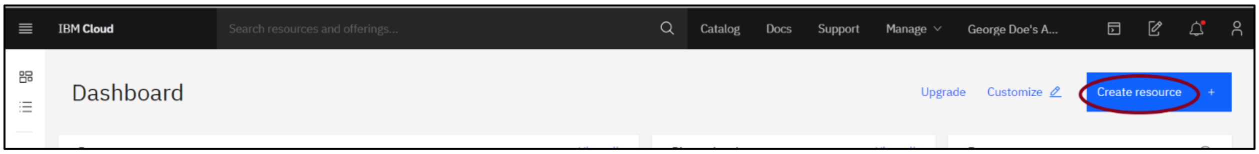
5. Click on **Dashboard**.



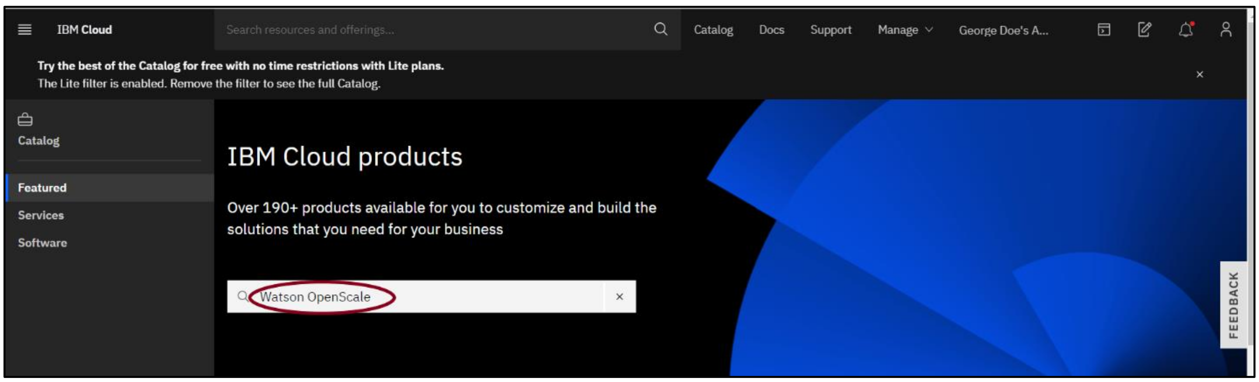
6. Note that the browser tab has been renamed to IBM Cloud.



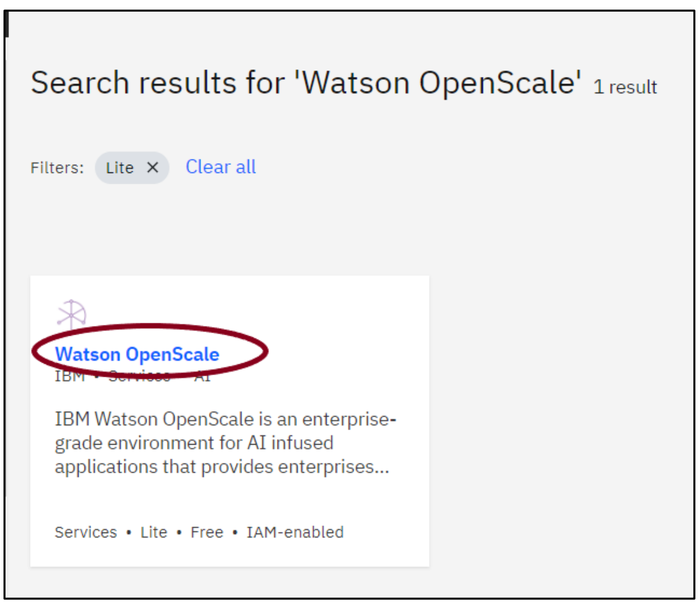
7. Click on **Create resource.** <u>If you have already completed the OpenScale setup, skip to step 14.</u>



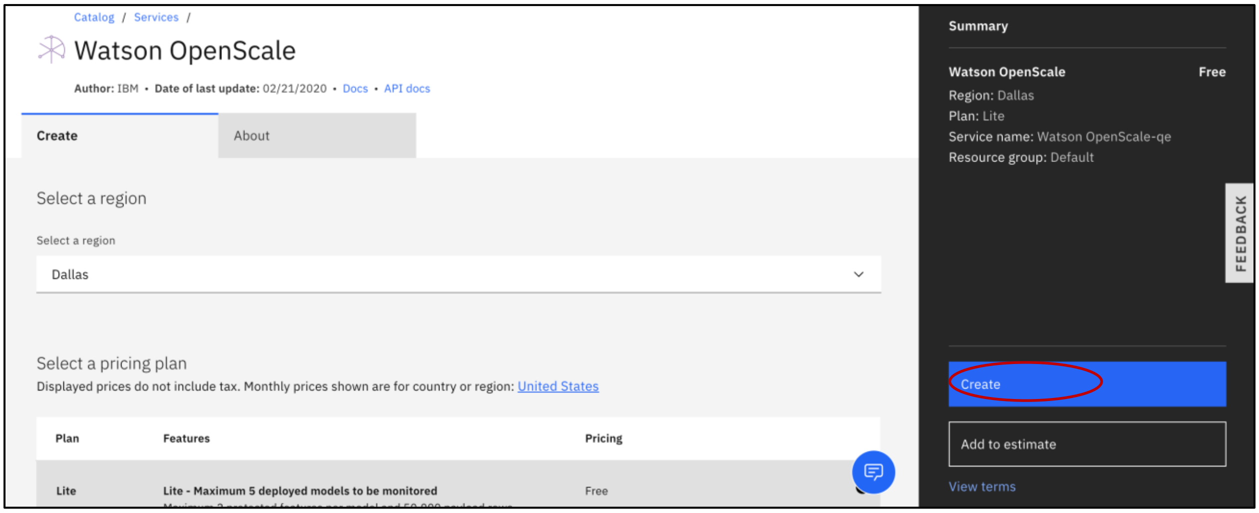
8. Enter **Watson OpenScale** and hit the **<Enter>** key.



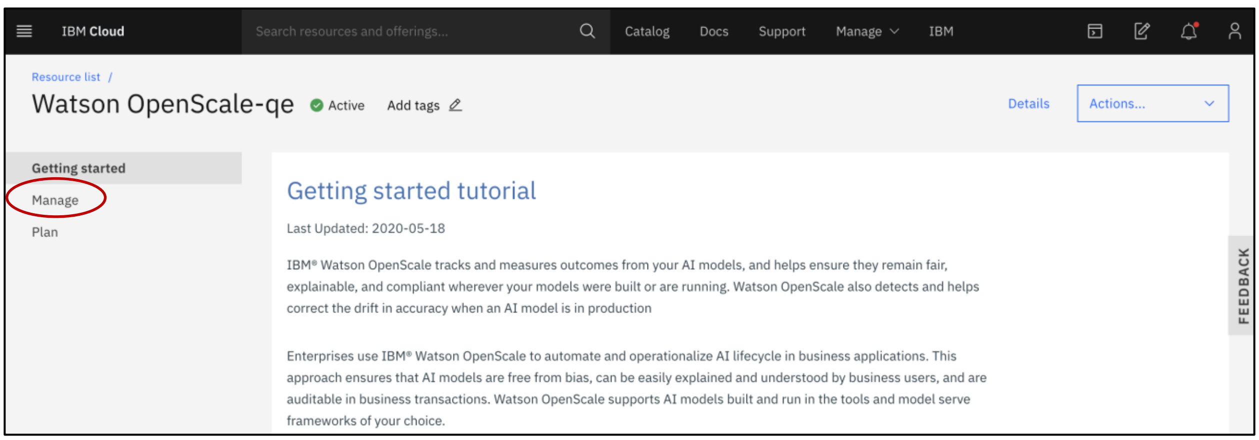
9. Click on **Watson OpenScale**.



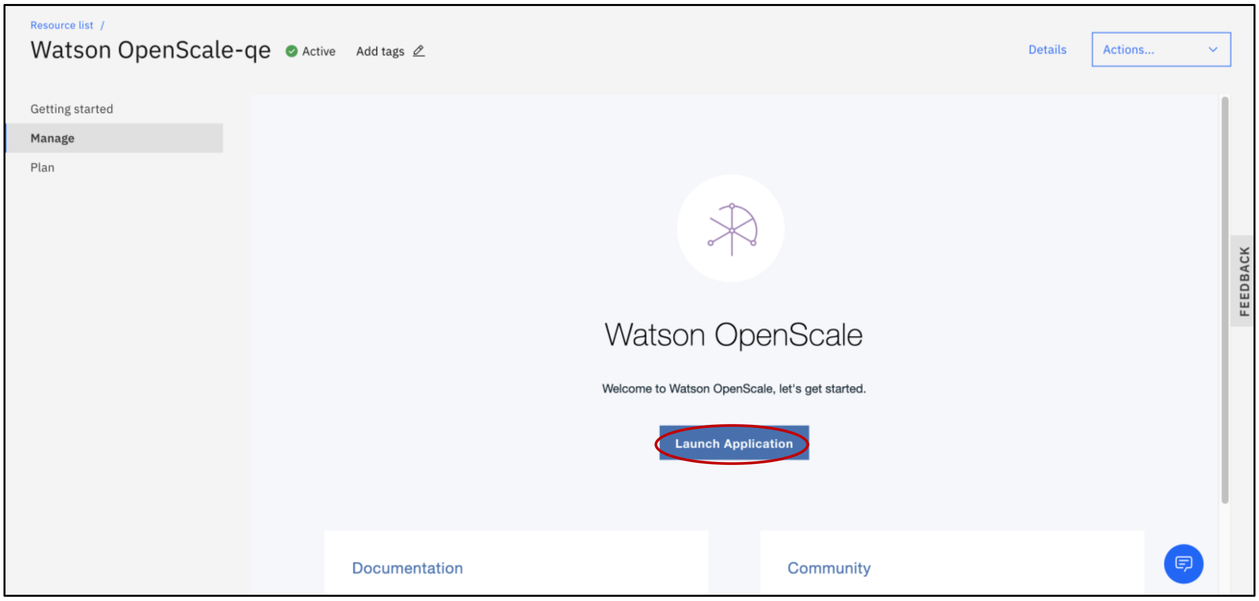
10. Click on **Create**.



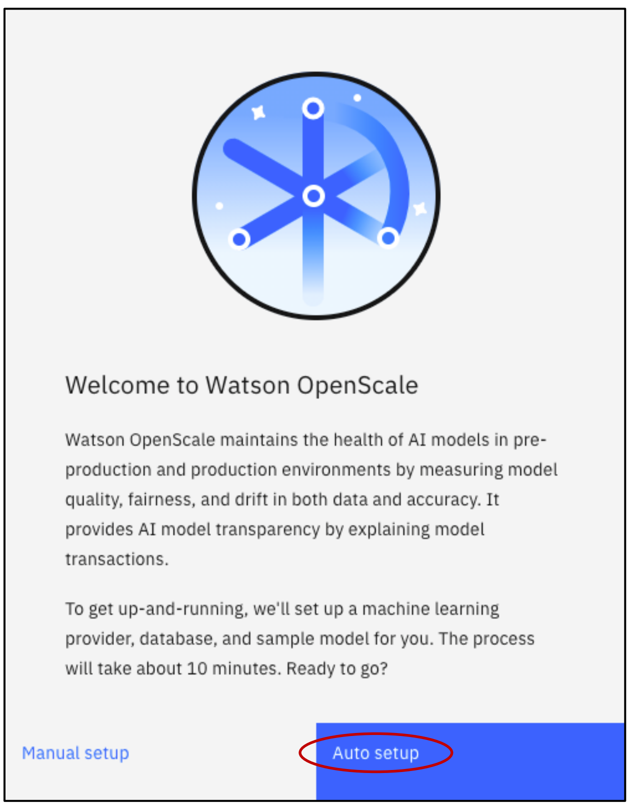
11. Click on **Manage**.



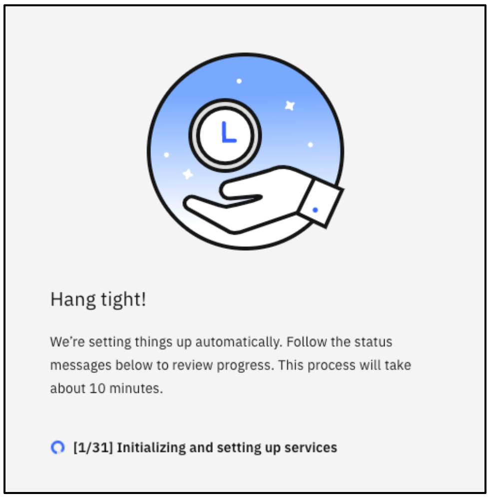
12. Click on **Launch Application**.



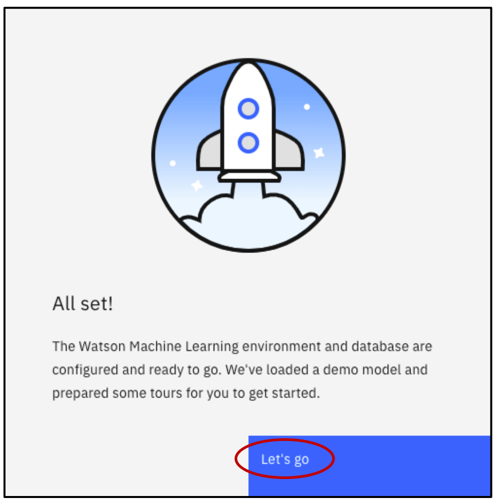
13. Make sure to click on **Auto setup.**



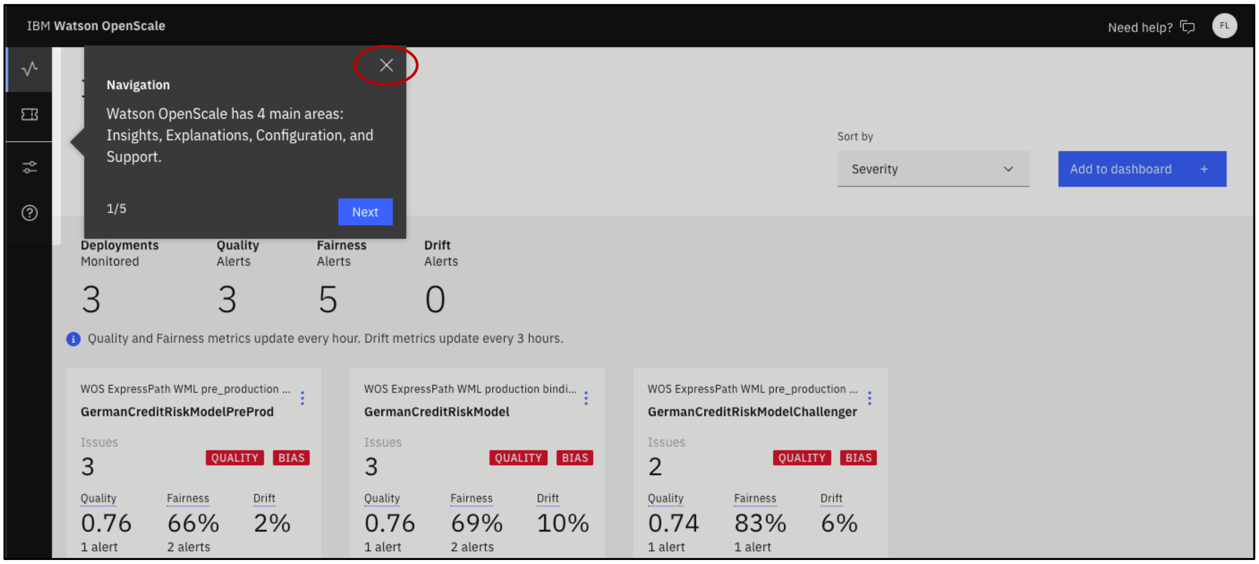
It will take about 10 minutes for this setup to complete. You will see the following screen as the setup runs:



14. Once the setup is complete, click **Let's go**.



15. Click on the **X** to close the series of tutorial popups.

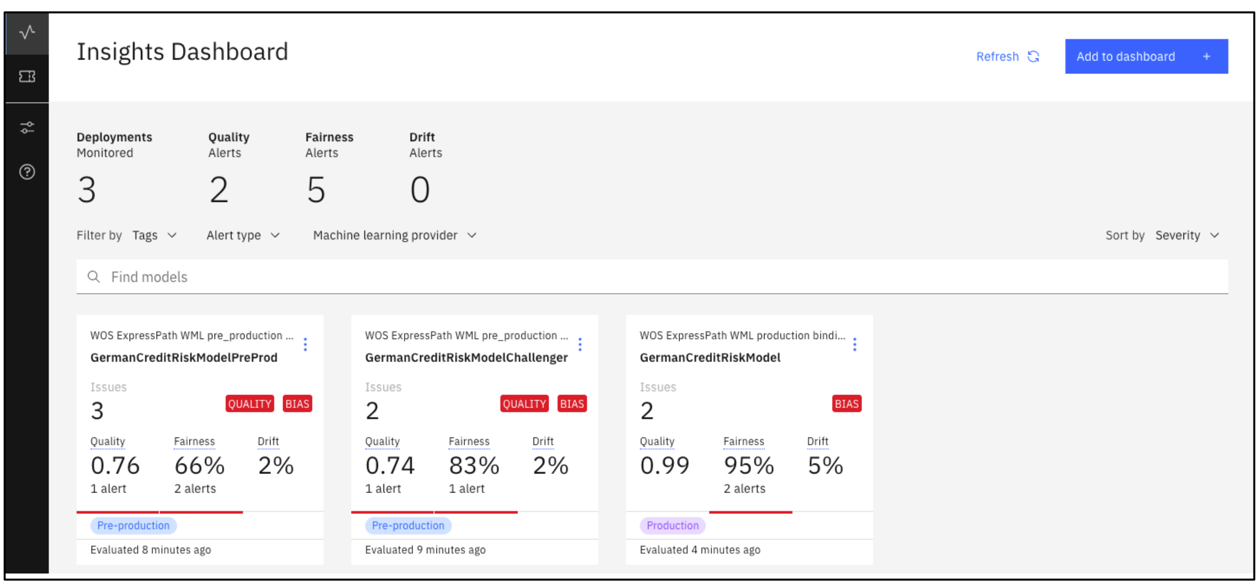


View Fairness and Quality Metrics

Watson OpenScale provides two types of monitors: application monitors, and model monitors.

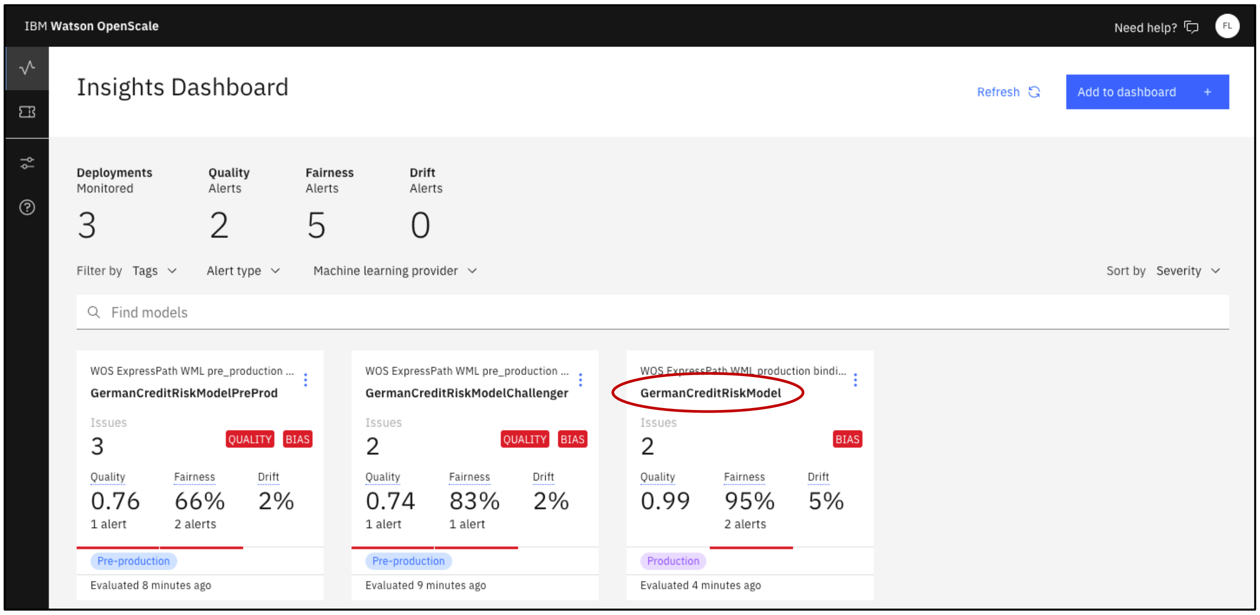
We will begin with the model monitors. The model monitors section of the Insights Dashboard provides an overview of all the models being monitored by OpenScale.

The dashboard shows how many deployments are currently being monitored, as well as an overview of the alerts from those models. Below, each model is represented by a tile showing the machine learning provider and alerts for that model.

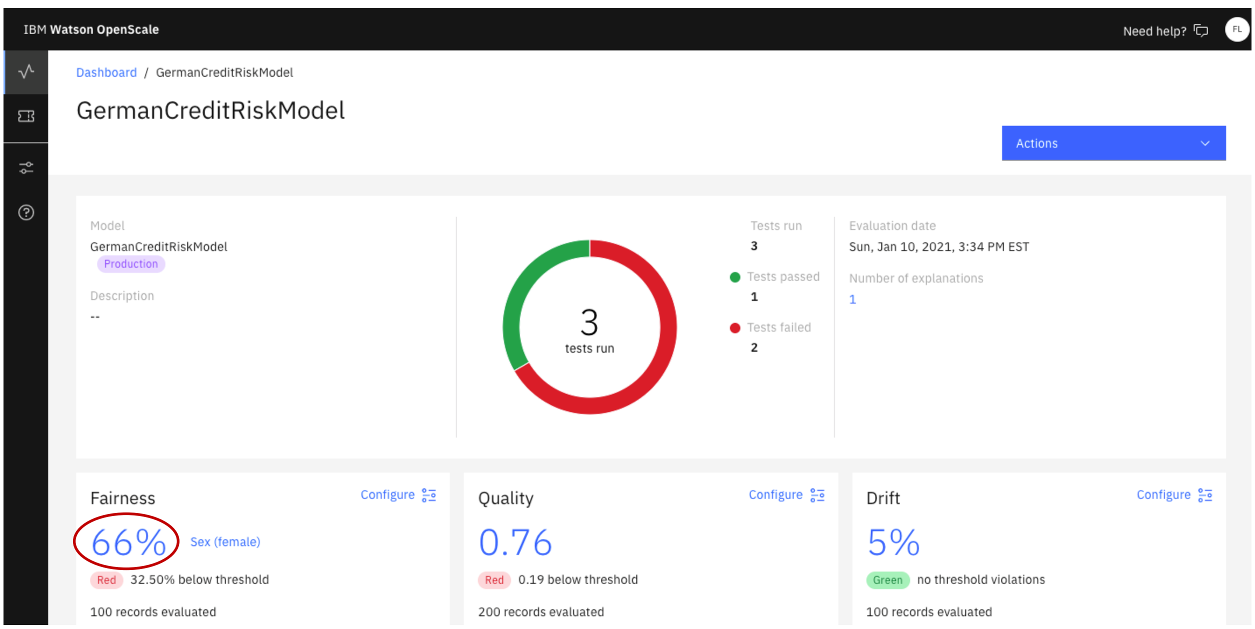


Note: Your actual values may vary slightly from what is shown in these lab screen shots as the data source has a level of randomness built into it.

1. Click on **GermanCreditRiskModel**

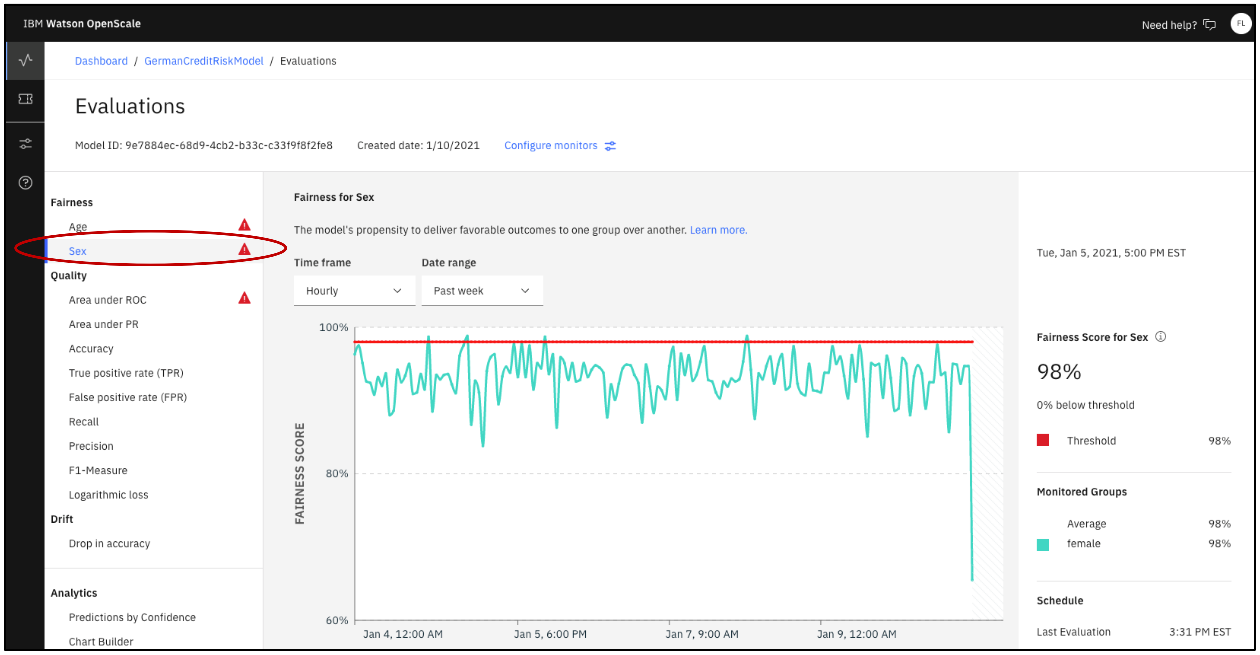


2. You are presented with a dashboard of your model evaluations. Click on the **percentage value under Fairness**.



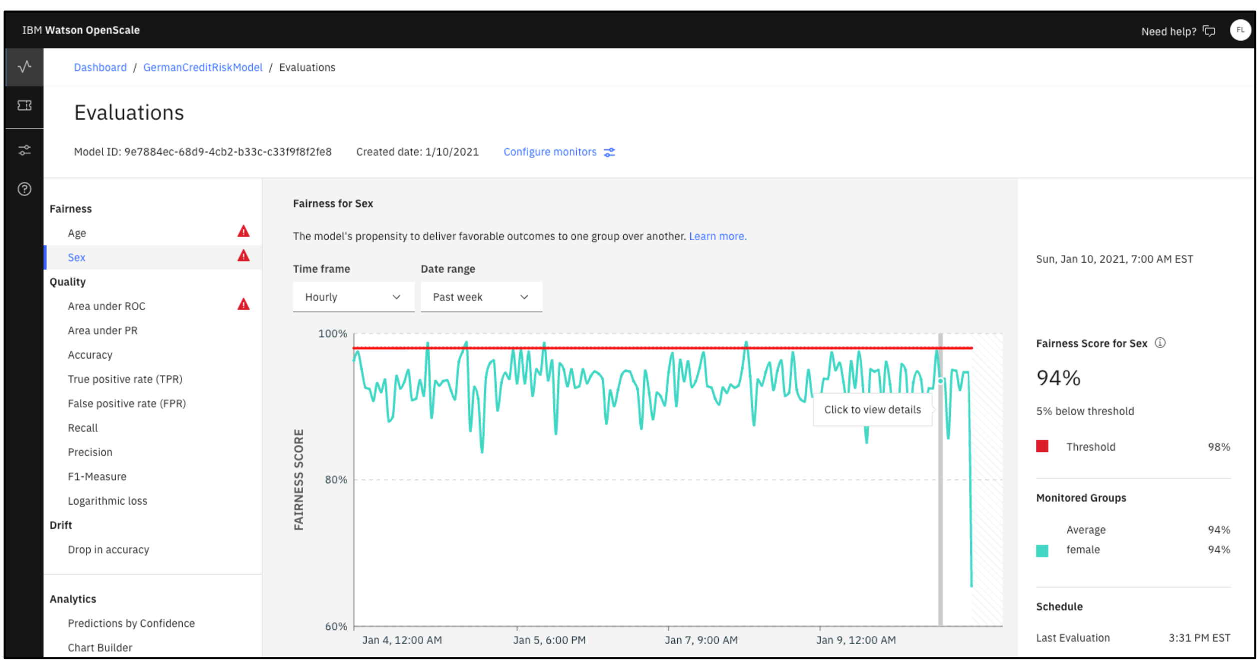
In the Fairness section, you can see that we have chosen to model two features, Age and Sex, for fairness. Additionally, you can see that we have an alert for the Sex feature.

3. Click on **Sex**.

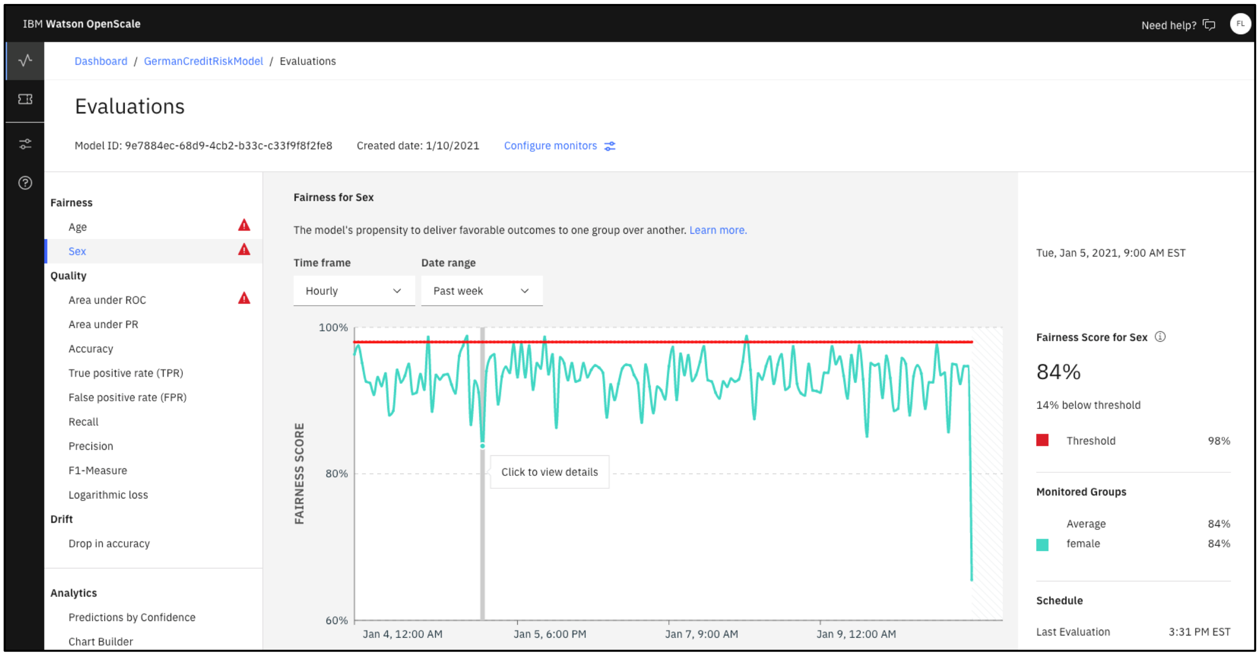


Note the time frame above the graph. We are looking at hourly data from the past week, but the time frame can be adjusted as necessary. The graph shows the fairness score for females as a green line. The threshold we have set for an alert is shown by the red line.

As you can see, the fairness score has dropped beneath the threshold consistently over the past week, alerting us to a potential unfair bias issue with the model. As you move your cursor inside that date range, values are updated to show the Fairness Score for that date and time.

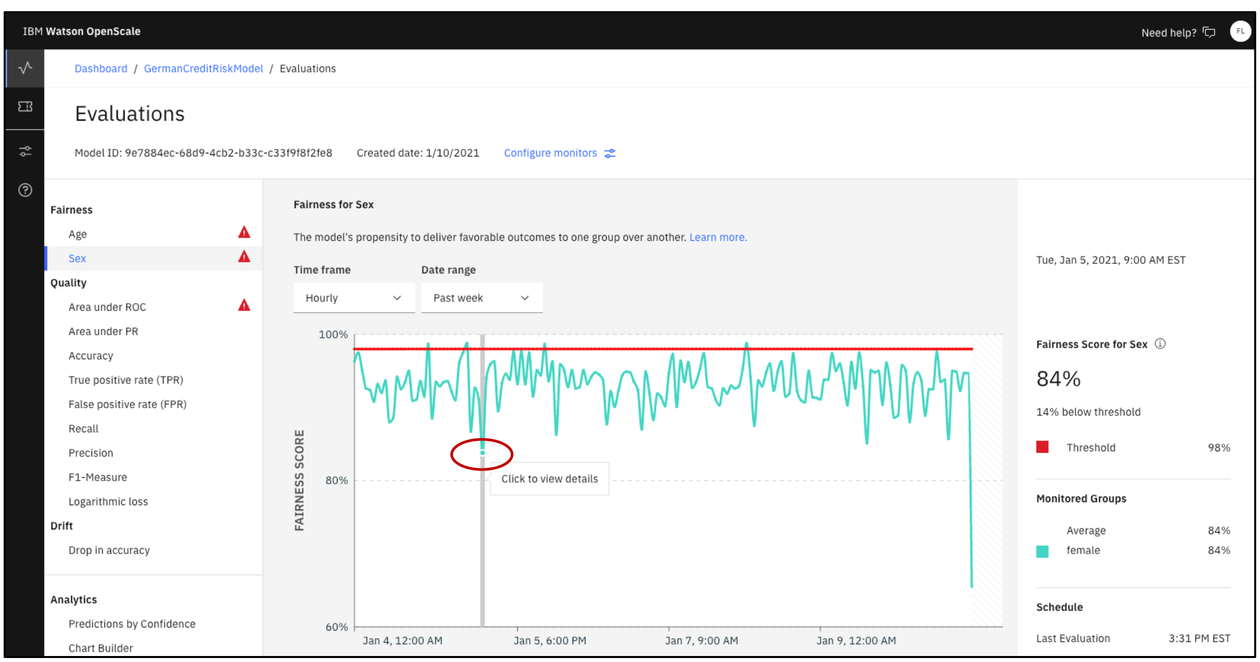


4. Move your mouse on the chart to the point where fairness score was at its lowest (84%)

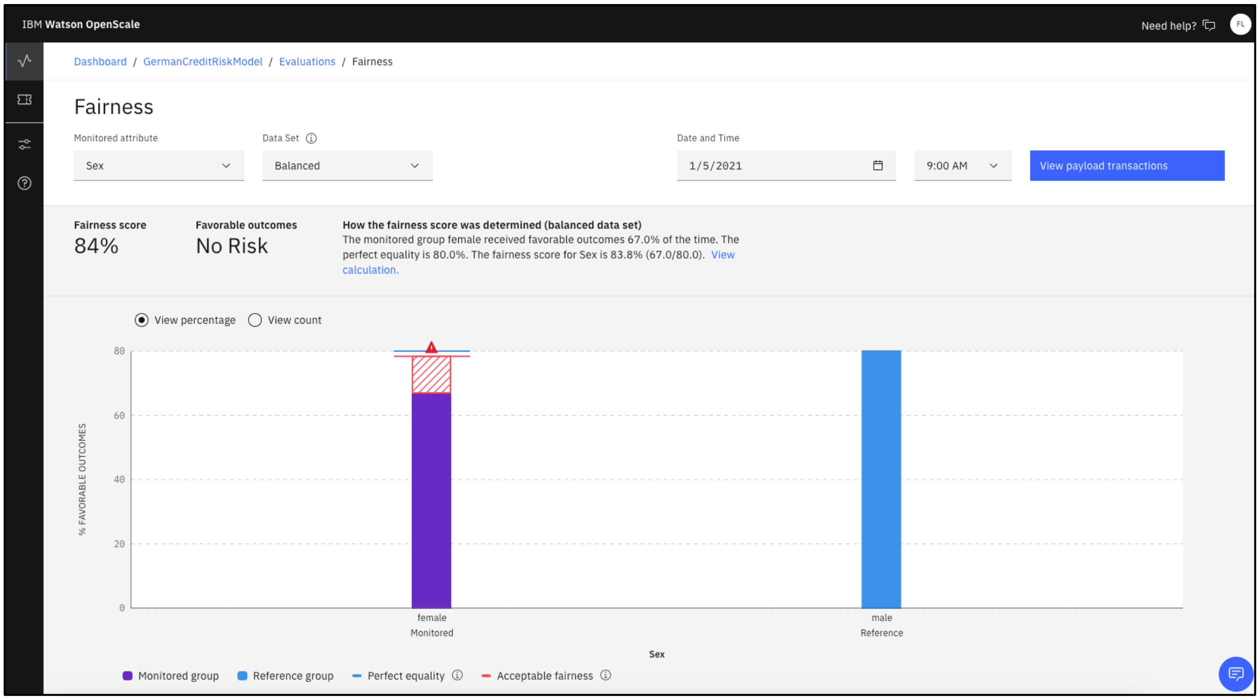


This screen shows us fairness details for this particular time period. OpenScale calculates its fairness score using a combination of actual predictions (payload data) and perturbed data, generated when the prediction probability is close to 50%. OpenScale will flip the monitored feature to see how it affects the prediction outcome.

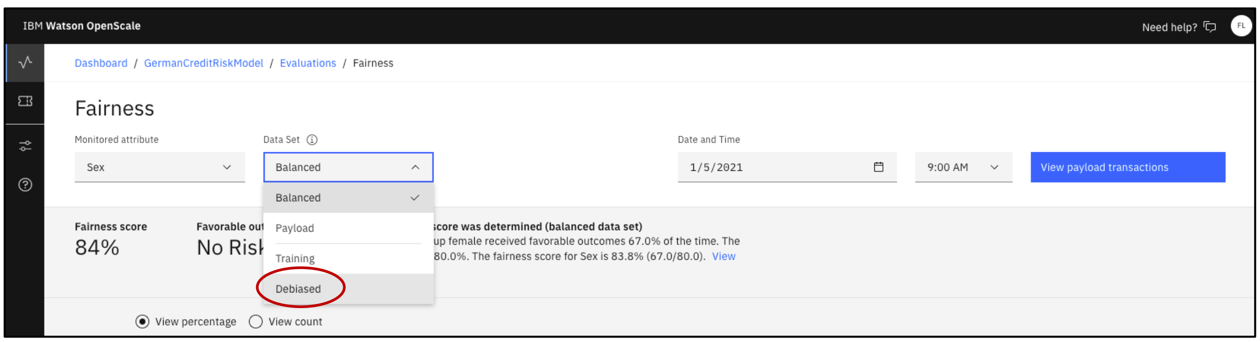
5. Click on that **lowest score line** when it says: Click to view details.



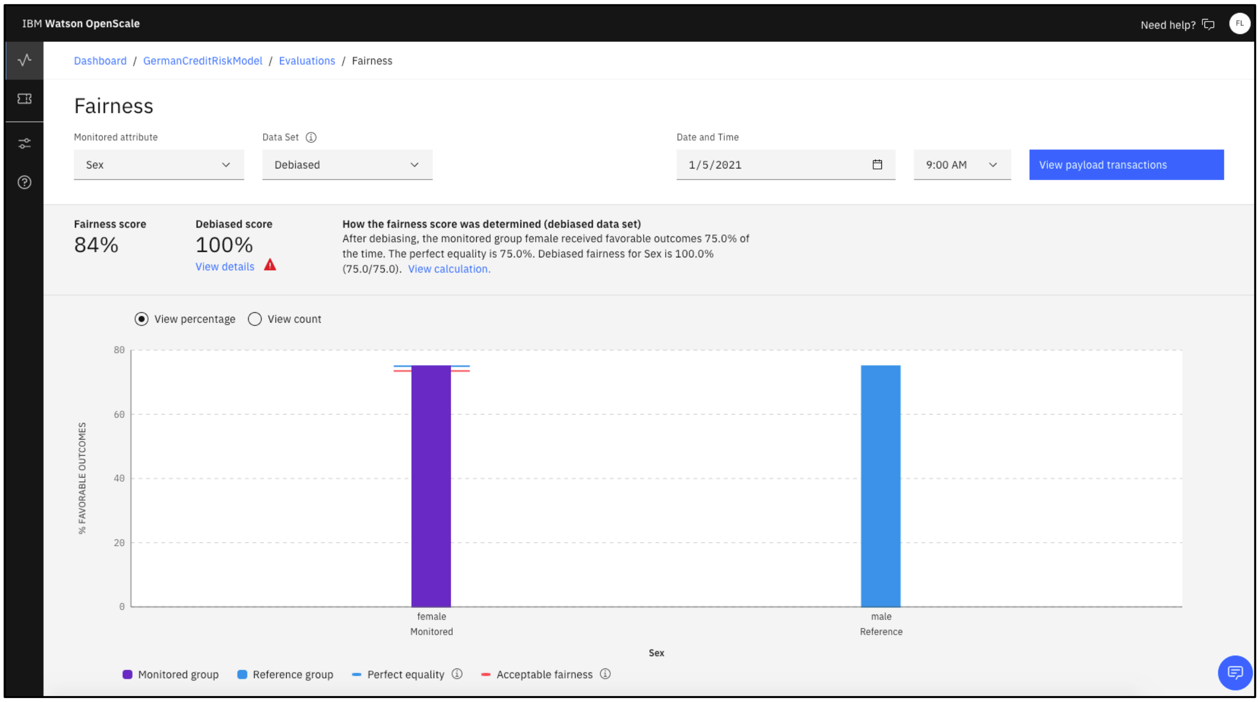
The fairness score is reached by dividing the percentage of positive outcomes for the monitored group (females, 67%) by the percentage of positive outcomes for the reference group (males, 80%). The graph shows the breakdown of positive and negative predictions for our two groups. You can use the Data Set dropdown at the top to view Balanced, Payload, Training, and Debiased data.



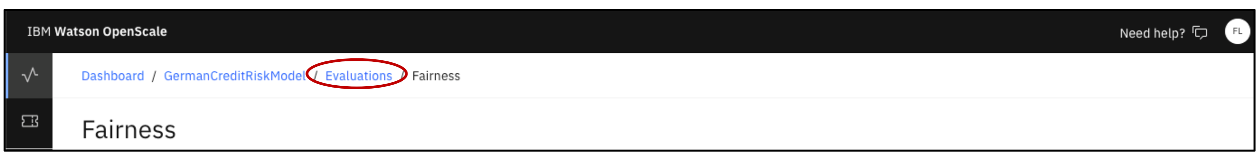
6. On the **Data Set dropdown**, select **Debiased**.



OpenScale can create a sort of "corrective lens" to reduce or remove unfair model bias. It does this by training another model to predict when an outcome of the production model is likely to be unfairly biased, flipping the feature value from the monitored group (female) to the reference group (male) and returning this prediction. On this screen, you can see how using this model will affect the fairness scores for other features.

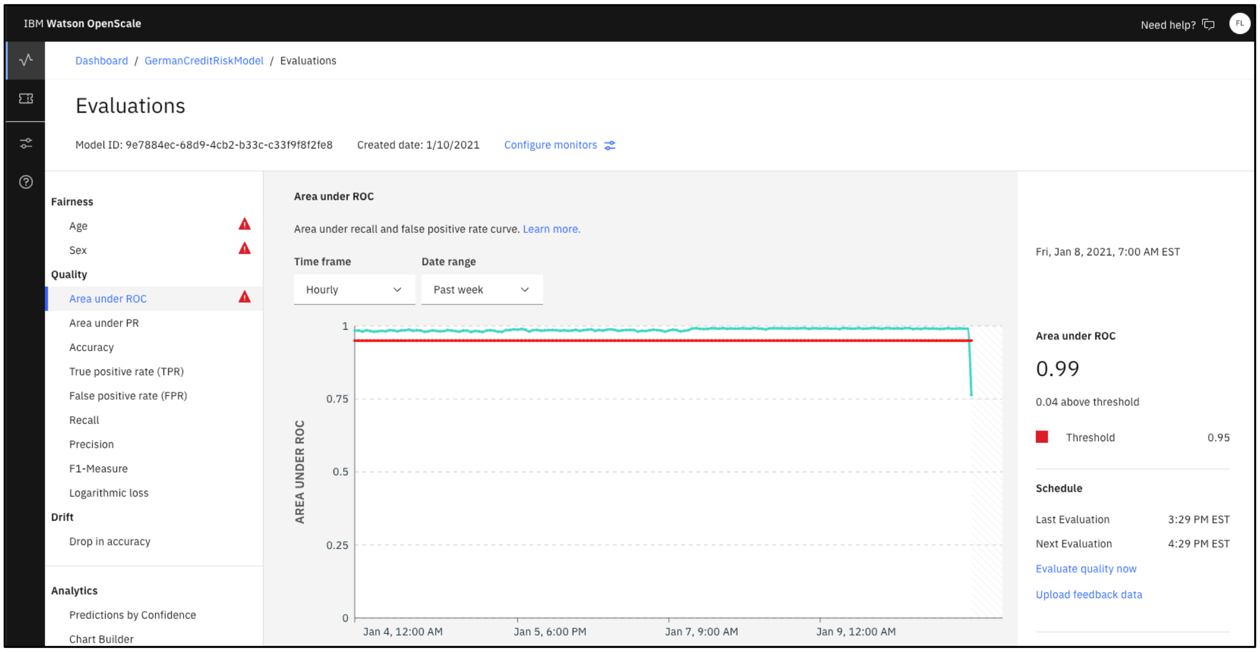


7. Click **Evaluations** to return to the model dashboard.



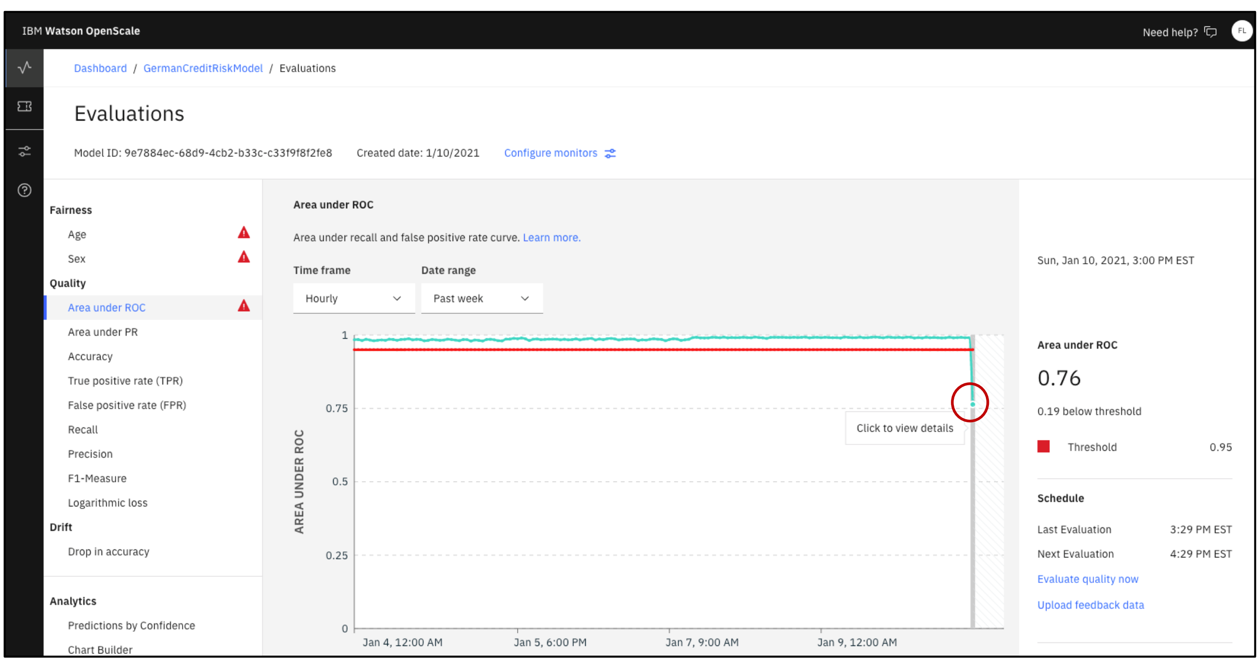
8. In the Quality section, click **Area under ROC**.

OpenScale provides several different quality measurements. For our binary classification model, Area under ROC provides the best standard for model quality. These scores are generated by providing ground truth feedback data to the model, either via CSV upload or using a RESTful endpoint provided by OpenScale.

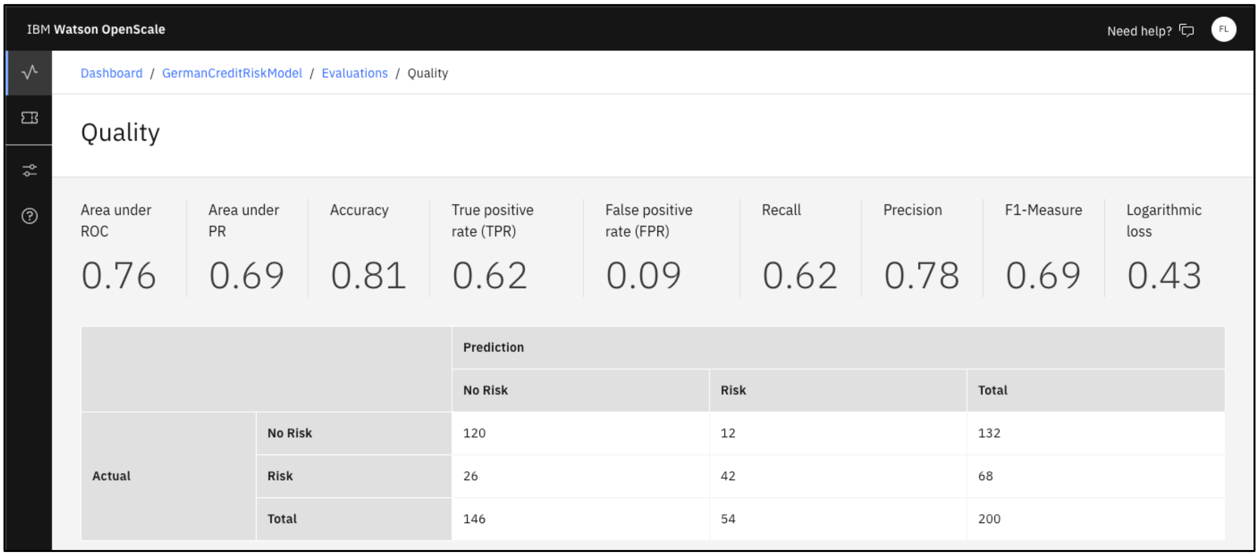


As with the fairness monitor, the chart in the middle of the screen shows model performance over an adjustable time window, with the relevant measurement shown as the green line and the alert threshold represented by the red line. As you can see, our model quality has consistently been above the threshold until the most recent measurement, represented by the right-most portion of the chart.

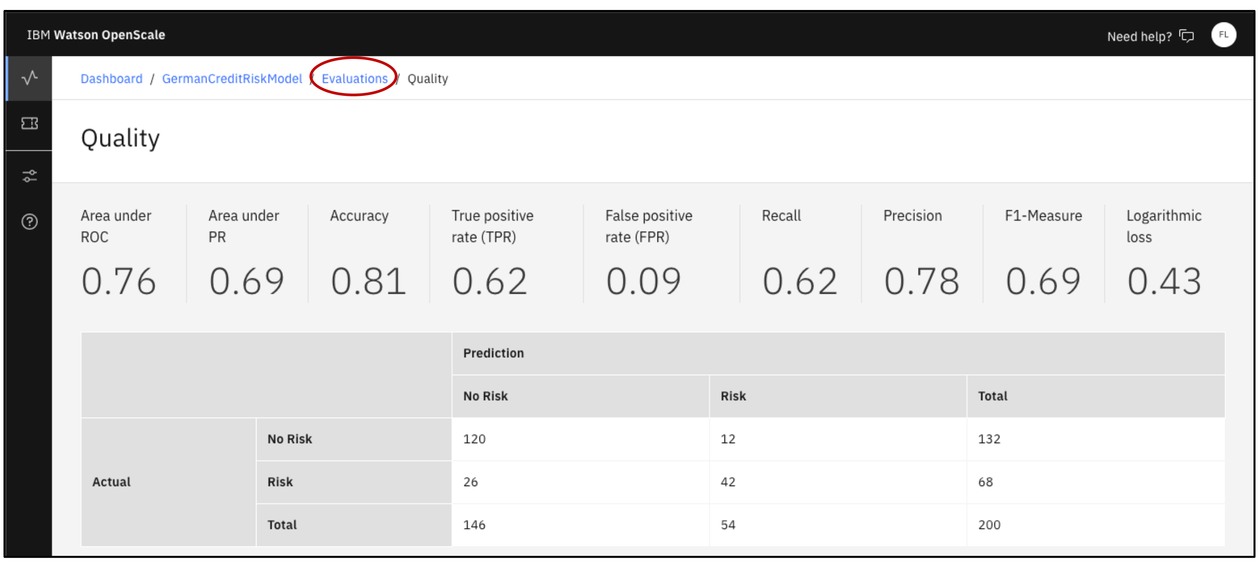
9. Click the right-most portion of the chart, where **Area under ROC drops to 0.76** or the lowest point on your chart.



Here we can see a further breakdown of the feedback data and the various accuracy scores, and the number of feedback records evaluated.

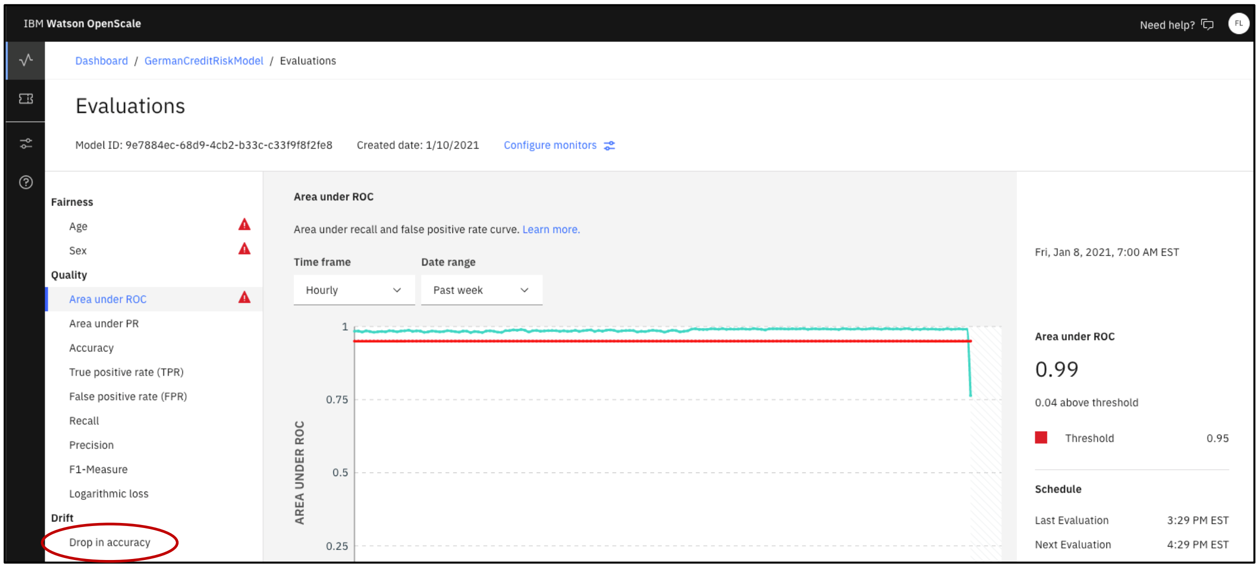


10. Click on **Evaluations** to return to the model dashboard.



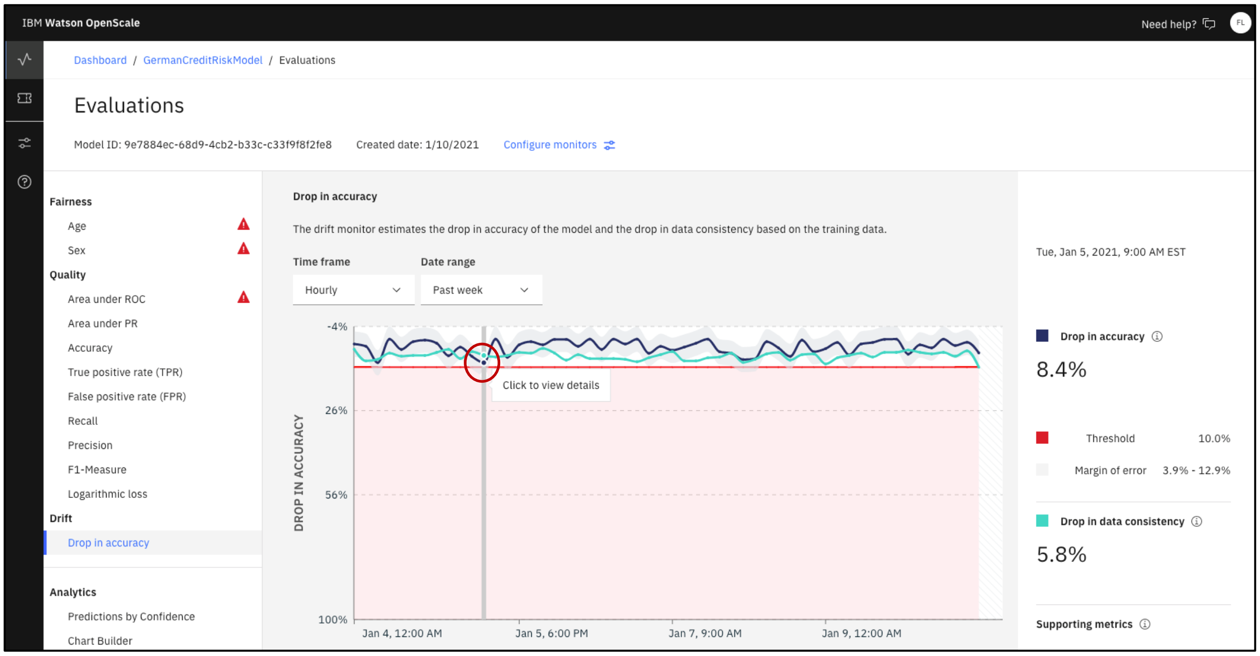
View Drift Metrics

1. Under Drift, click on **Drop in accuracy**.

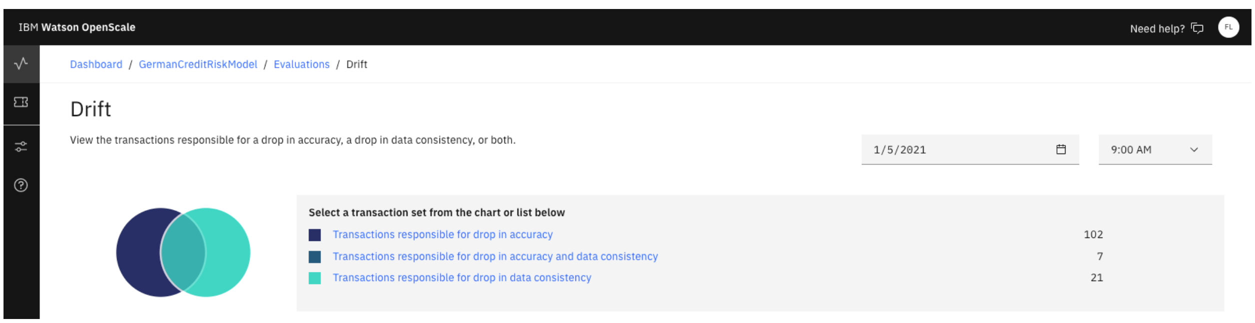


The OpenScale drift monitor is a separate linear regression drift model, trained to determine which types of data the production model struggles to correctly predict. This drift model allows OpenScale to forecast potentially costly drops in model accuracy without requiring additional feedback data. Additionally, the drift monitor compares incoming prediction requests with the training data to identify changes in data consistency that also may affect model output.
These two measurements are shown on the drift monitor screen. Estimated drop in accuracy is represented by the black line, drop in data consistency by the green line, and alert threshold by the red line.

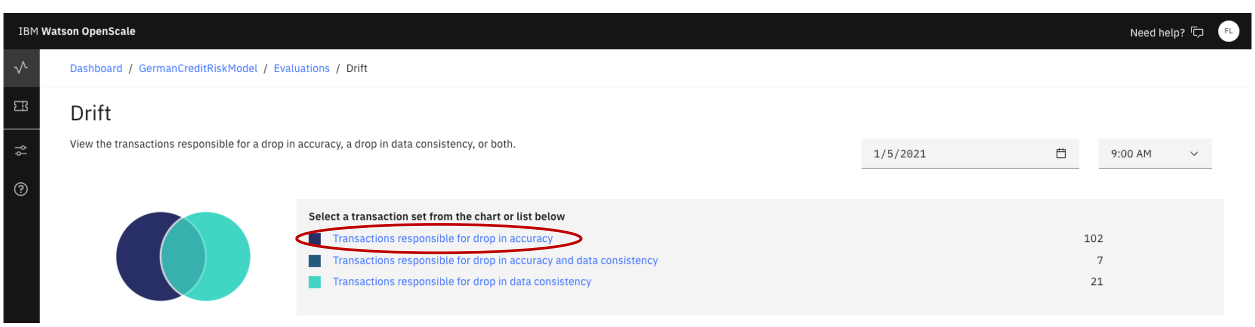
2. Click where drop in accuracy is greatest (8.4%).



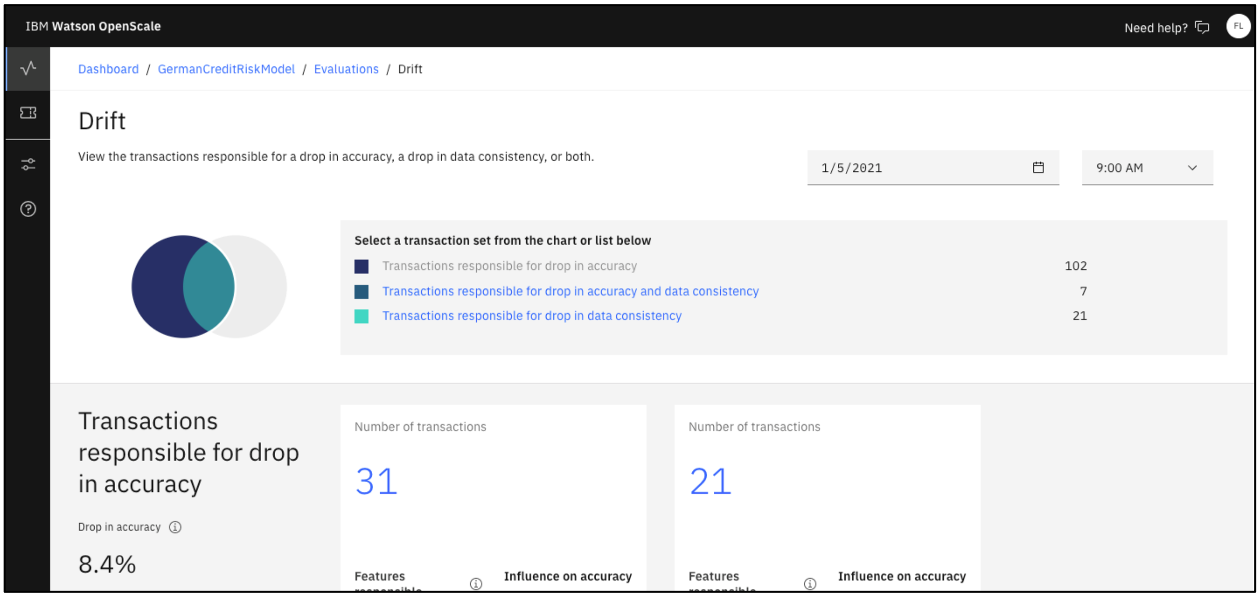
Here we can get a detailed view of the transactions responsible for estimated drops in accuracy, data consistency, or both.



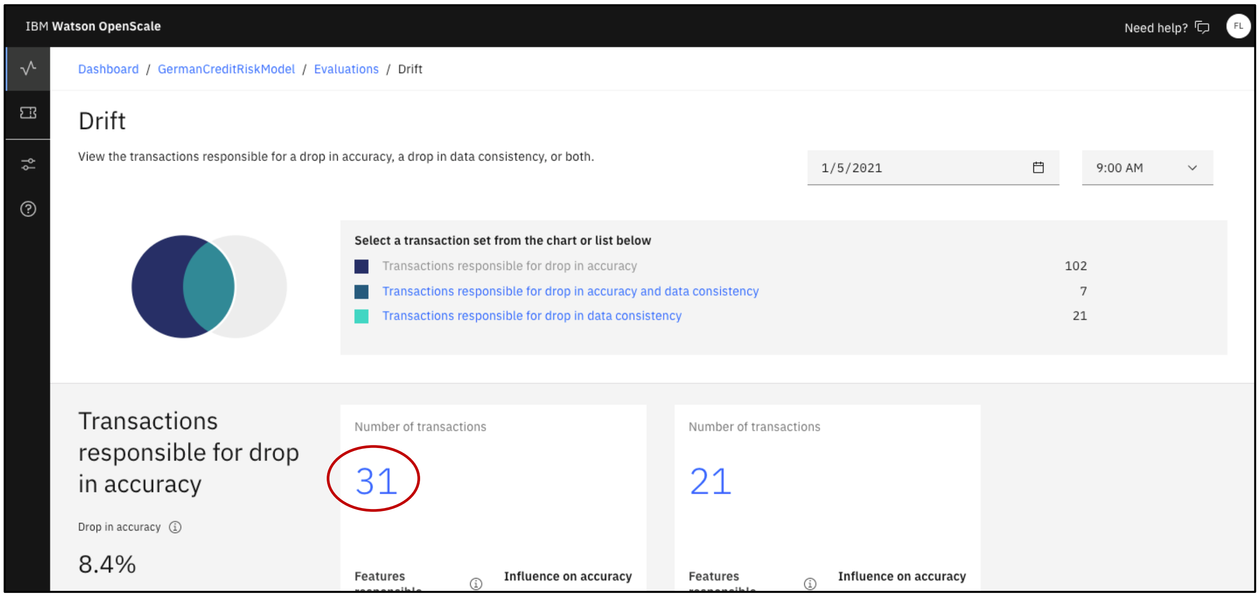
3. Click **Transactions responsible for drop in accuracy**.



OpenScale divides transactions (predictions) that are affecting model accuracy into groups that share feature characteristics, providing a view of which feature values are causing our drift and how much influence each is having.

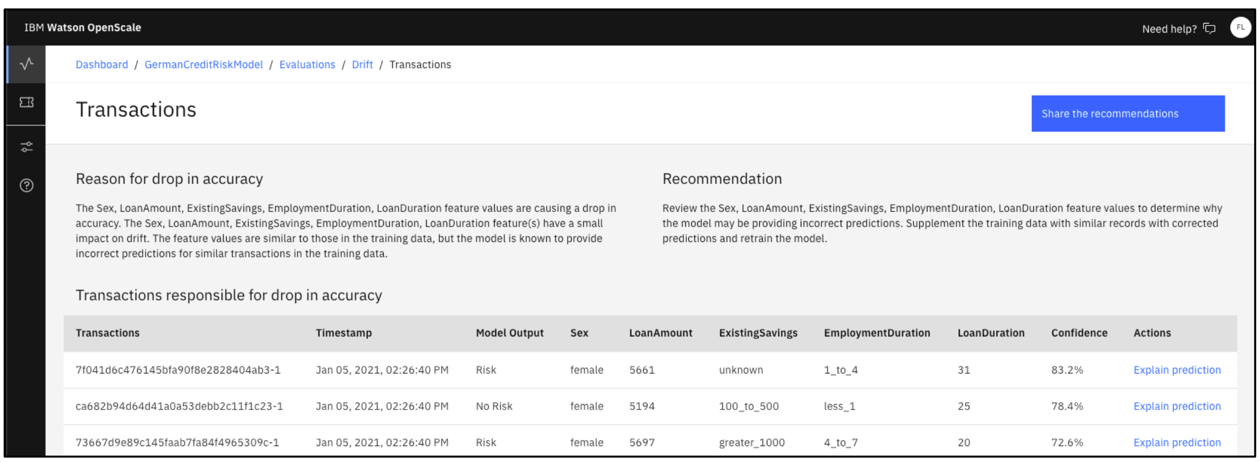

IBM Watson OpenScale
Need help?
FL
Dashboard / GermanCreditRiskModel / Evaluations / Drift
Drift
View the transactions responsible for a drop in accuracy, a drop in data consistency, or both.
1/5/2021
9:00 AM
Select a transaction set from the chart or list below
Transactions responsible for drop in accuracy 102
Transactions responsible for drop in accuracy and data consistency 7
Transactions responsible for drop in data consistency 21
Transactions responsible for drop in accuracy
Drop in accuracy
8.4%
Number of transactions
31
Features
Influence on accuracy
Number of transactions
21
Features
Influence on accuracy

4. Click on one of the **tiles** for a transaction grouping.



Here, OpenScale provides a detailed summary of how the values are affecting the model, as well as recommendations for how to address the issues with corrected training data.

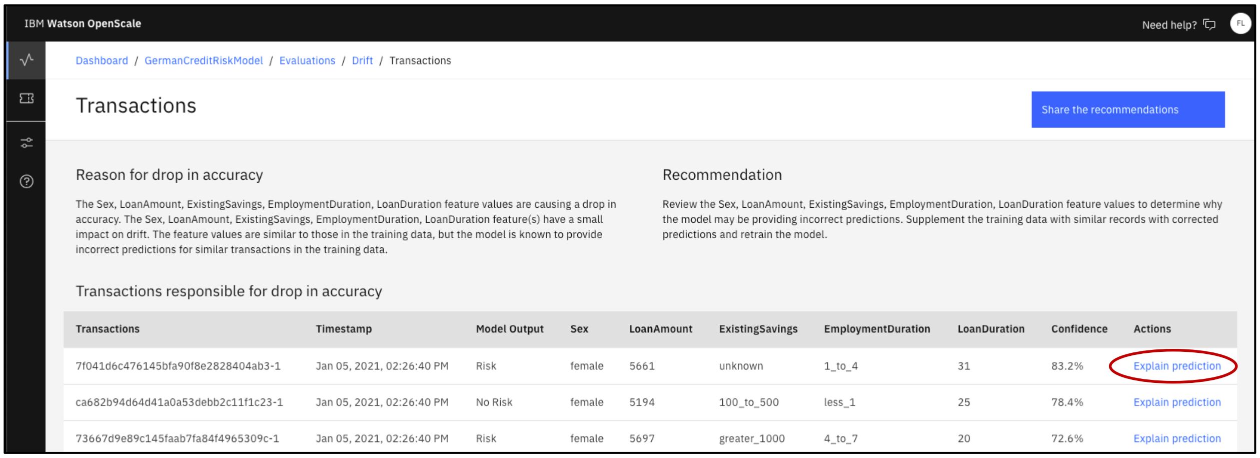
OpenScale lists the predictions and feature values for this grouping.



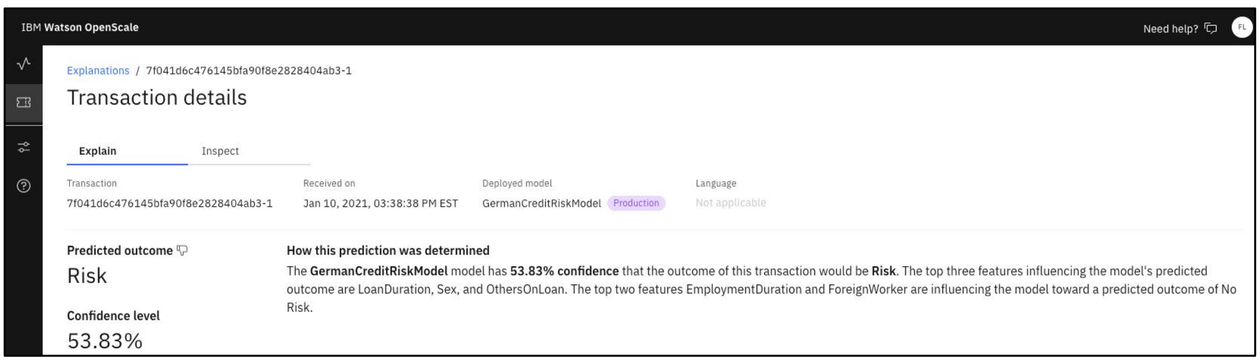
Explain a Transaction

Using a variety of open source algorithms, OpenScale can provide highly detailed explanations of the predictions your model has made.

1. Click an **Explain prediction** link under the Actions column.



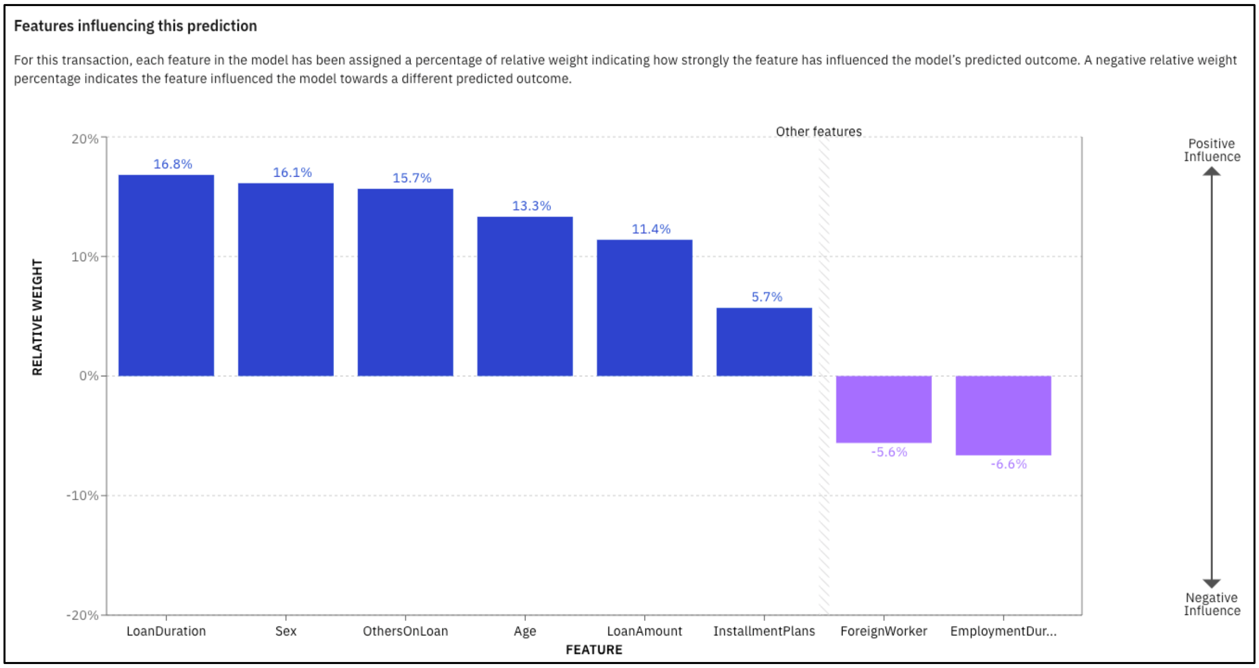
The OpenScale explanation feature works by slightly perturbing the feature values from the original prediction, sending these values to the production model, and measuring the impact the changes have on the outcome. By sending thousands of perturbed requests, OpenScale can gain a detailed picture of feature importance for not only relatively simple models like linear regression or decision tree classifiers, but also complex neural networks and image recognition models. **(It can take a minute or two for the details to be calculated, so please be patient.)**



The upper portion of the screen shows information on the model and original prediction, as well as IBM's contrastive explanation technology. The Minimum changes for No Risk outcome show the Pertinent Negative values, or the least amount the feature values for the transaction can be changed to get a different outcome.

The Maximum changes allowed for the same outcome show the Pertinent Positive values, or how much the feature values can change and still have the model make the same prediction.

2. **Scroll down** to see the factors that influenced the prediction.



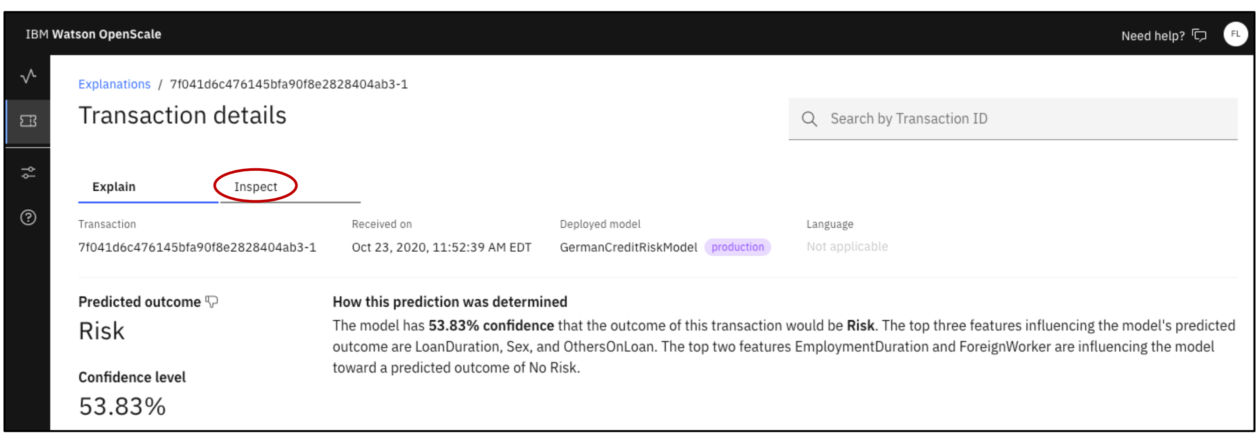
Here, you can see the confidence the model has in its prediction as well as a quick summary of the most important factors that led the model to make a prediction that this particular loan application represents a risky one.

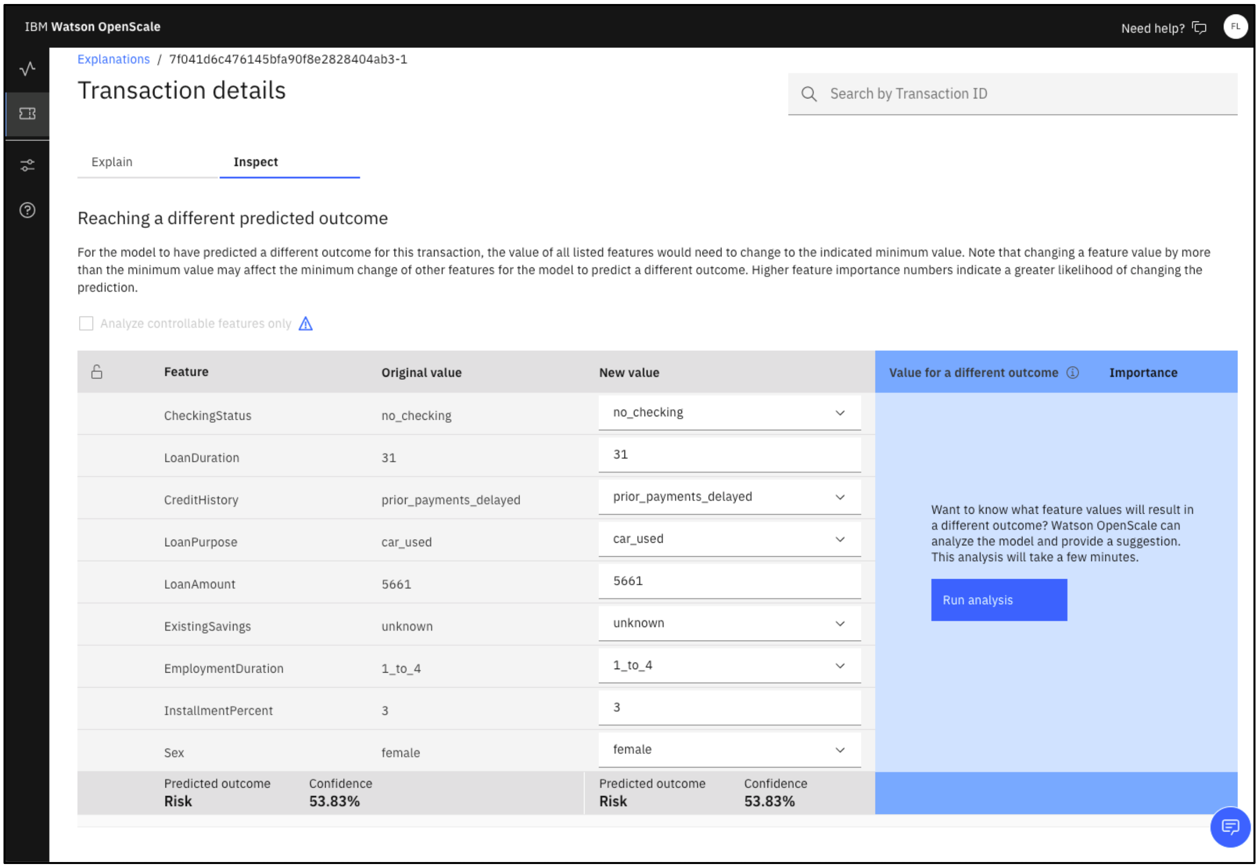
The chart at the bottom of the screen shows the feature values for this prediction, whether they contributed to a No Risk or Risk prediction, and how much they influenced the model.

This detailed information allows you to ensure that your models are making predictions grounded in reality, as well as providing full explainability for predictions in case of an external audit or internal review of the model.

Finally, the data from this explanation is stored in the OpenScale data mart, where it, along with all other metrics, can be retrieved via API and surfaced to business users or even customers via dashboards or other applications.

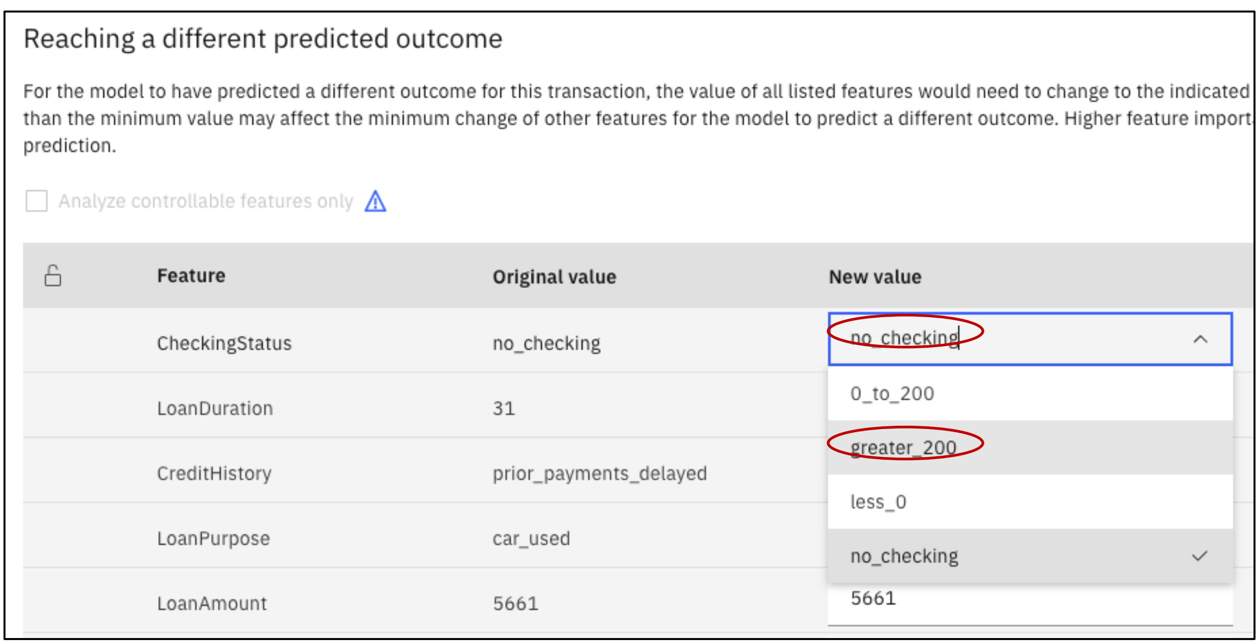
3. Click on **Inspect**



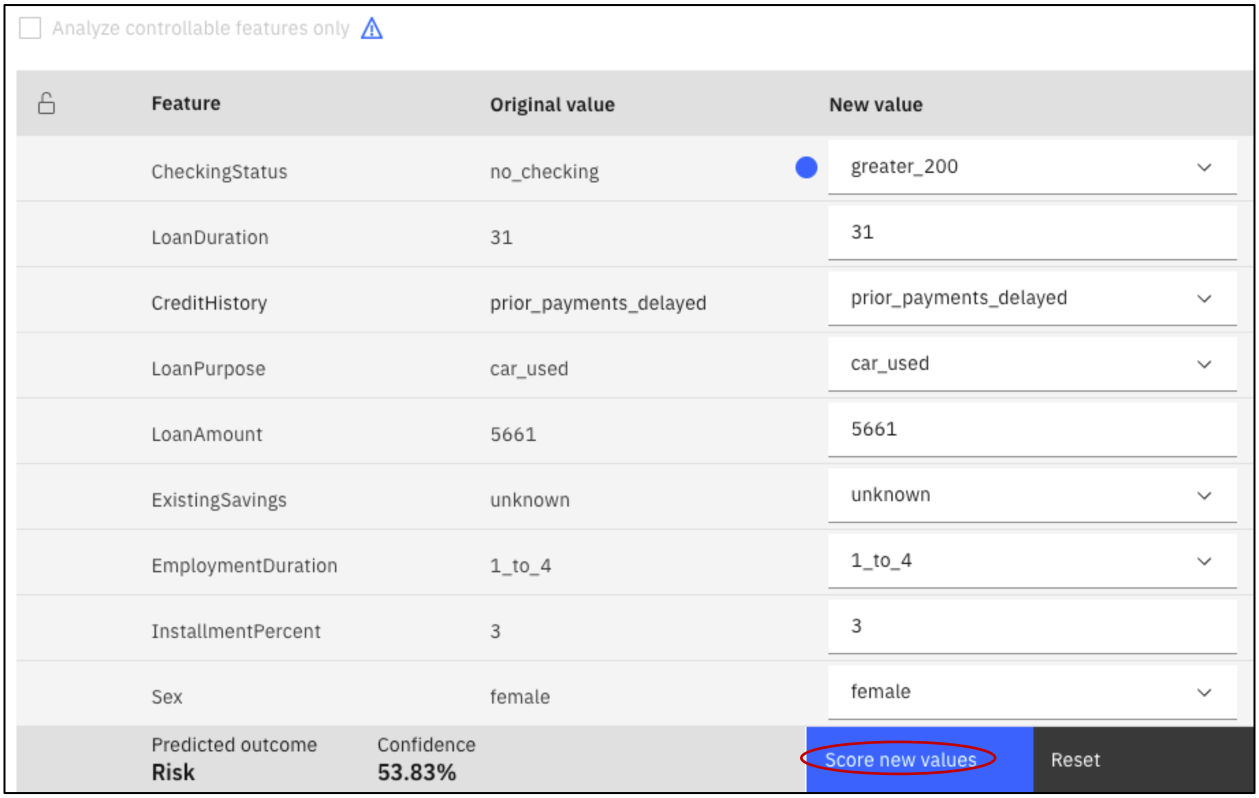


This tab allows you to enter specific values for each feature to test how the outcome would change.

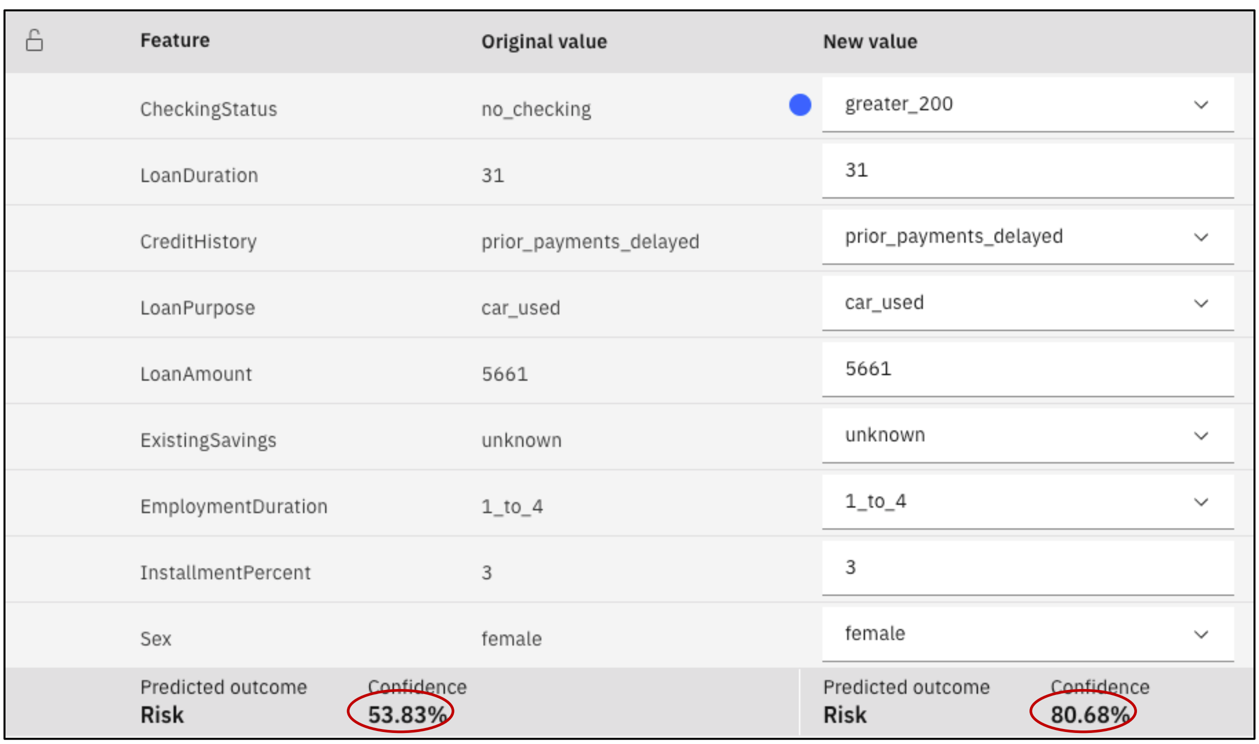
4. Change **checkingStatus** from no_checking to **greater_200**.



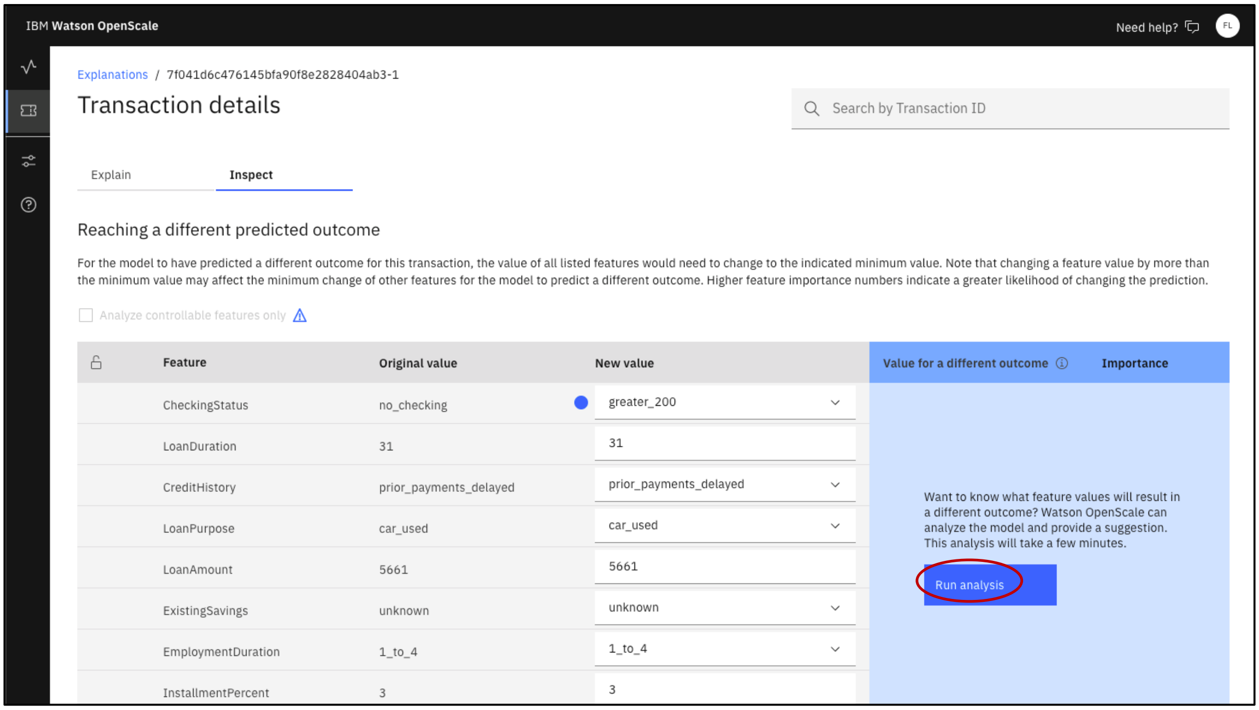
5. Click **Score new values**.



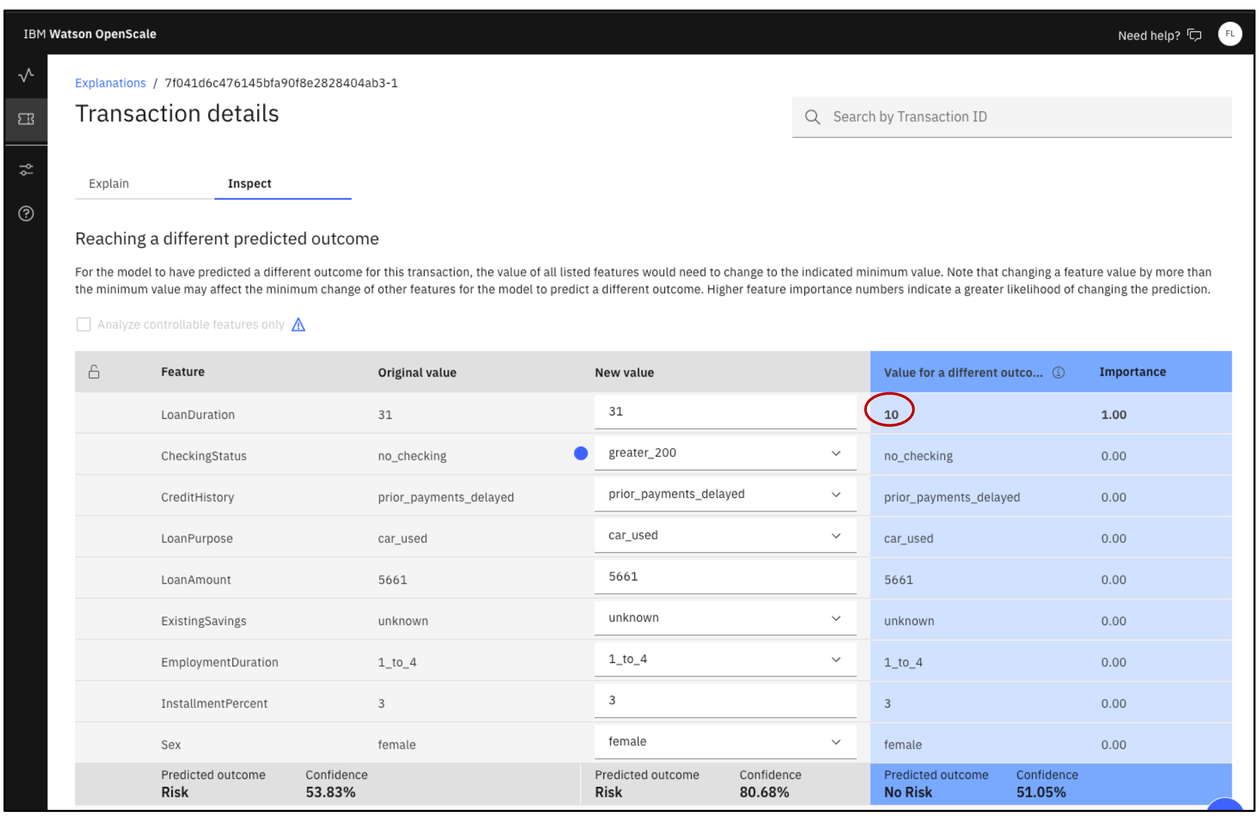
Note the change in Confidence.



6. Click on **Run analysis**.



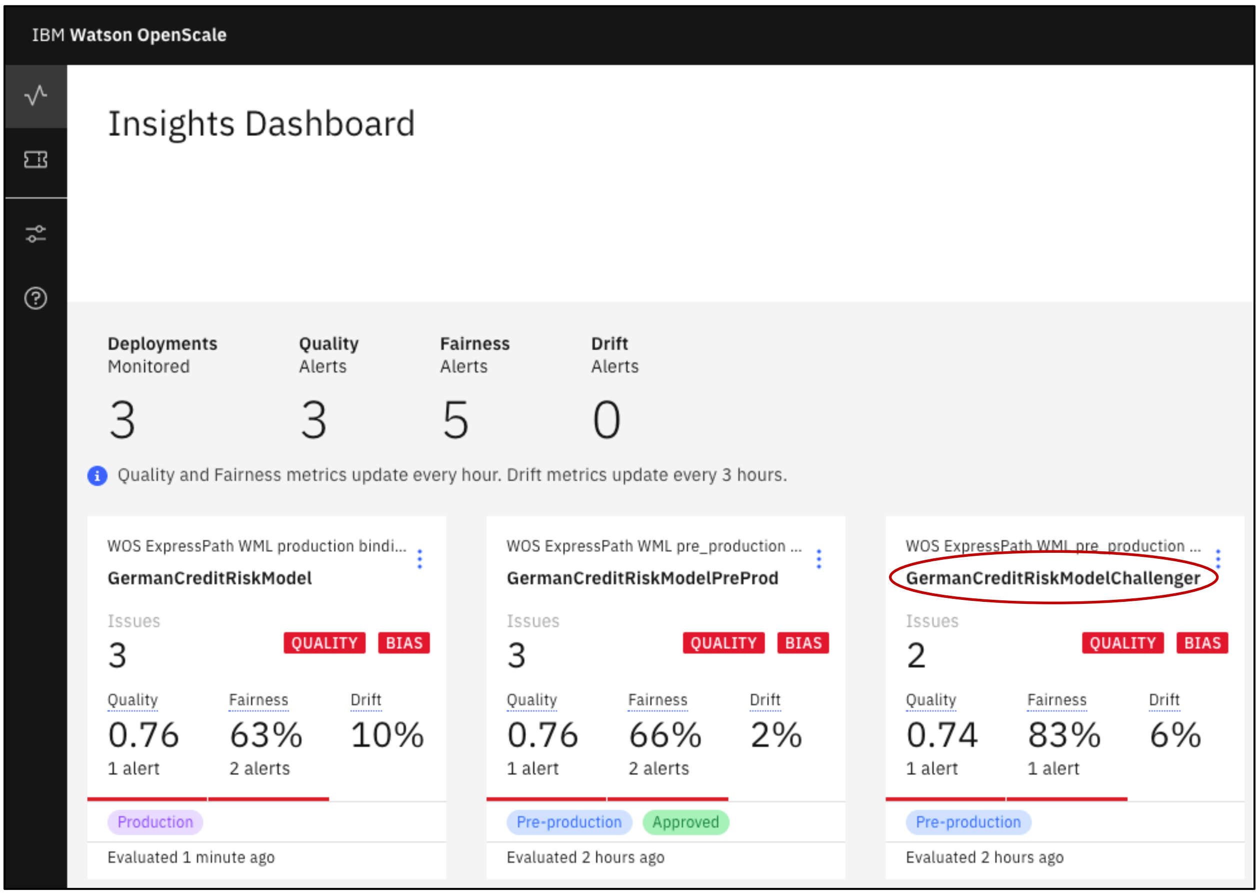
After OpenScale completes analysis, it displays the smallest change that can be made for a different outcome. Here it prioritizes changing only one feature.



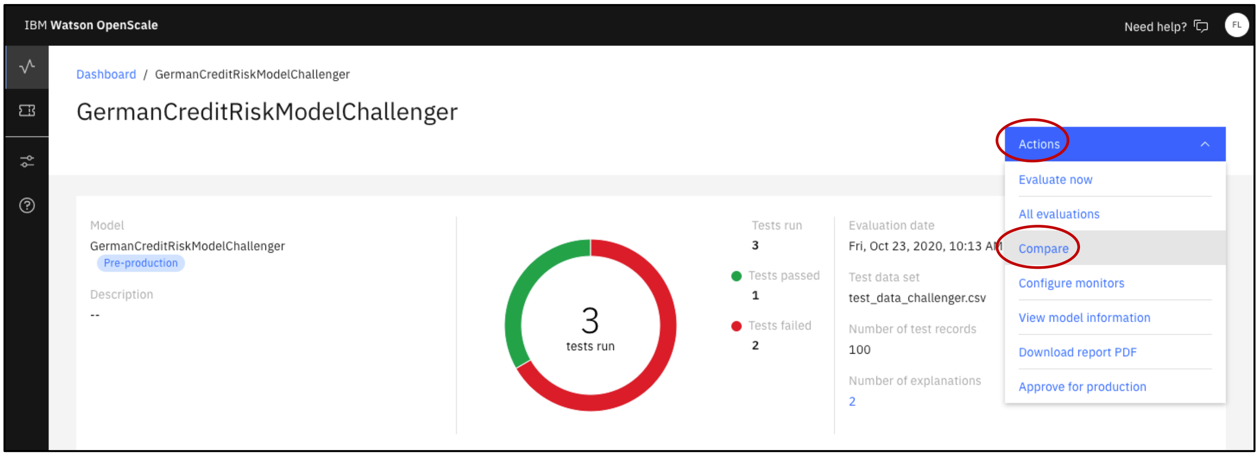
7. Click on the **monitor tile** to return to the main deployments overview dashboard.

Compare a Pre-production Model and Generate a Report

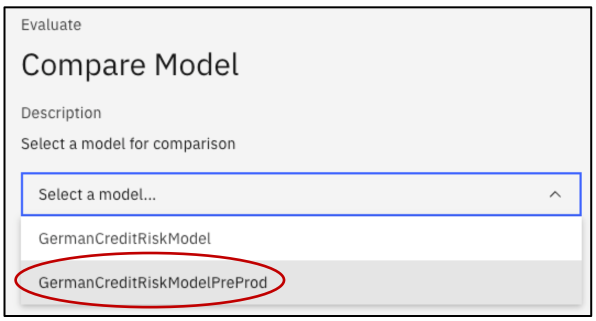
1. To compare how the pre-production models are performing, click on the **GermanCreditRiskModelChallenger** tile.



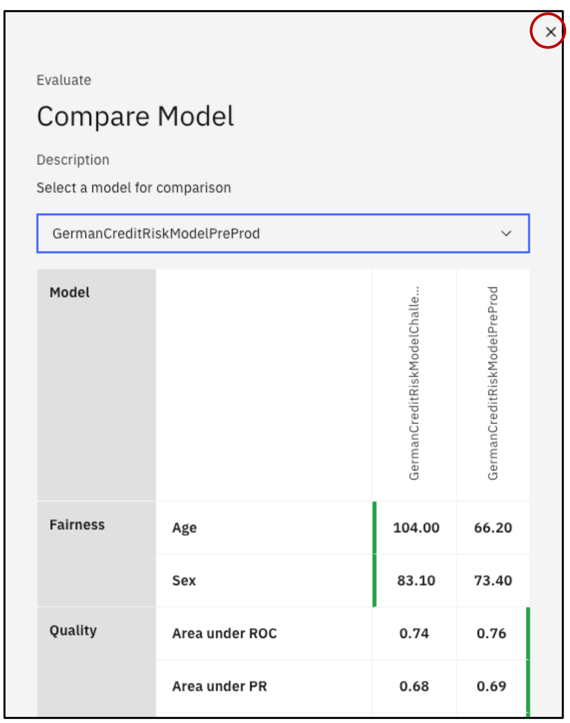
2. Select **Actions** and **Compare**.



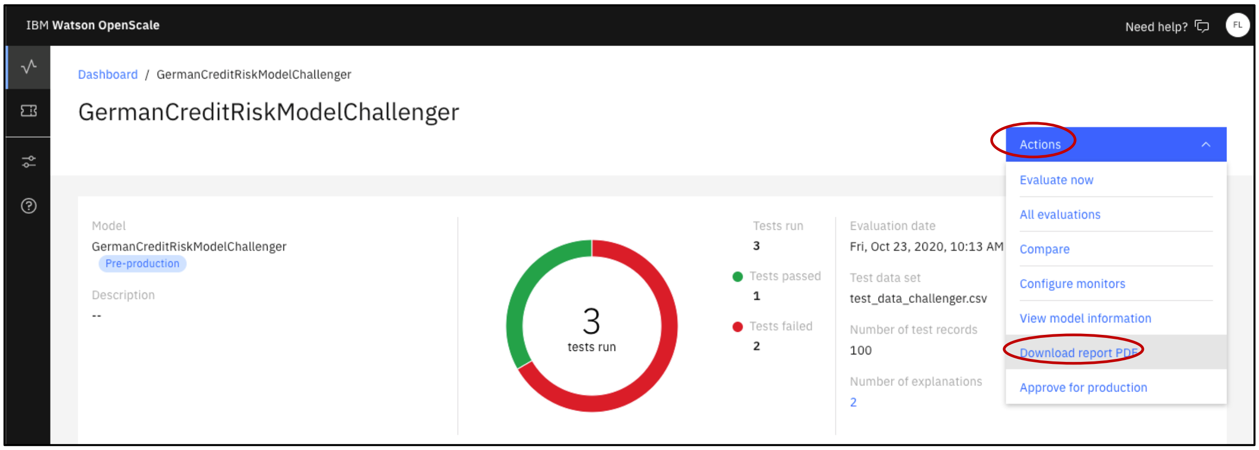
3. Select **GermanCreditRiskModelPreProd**



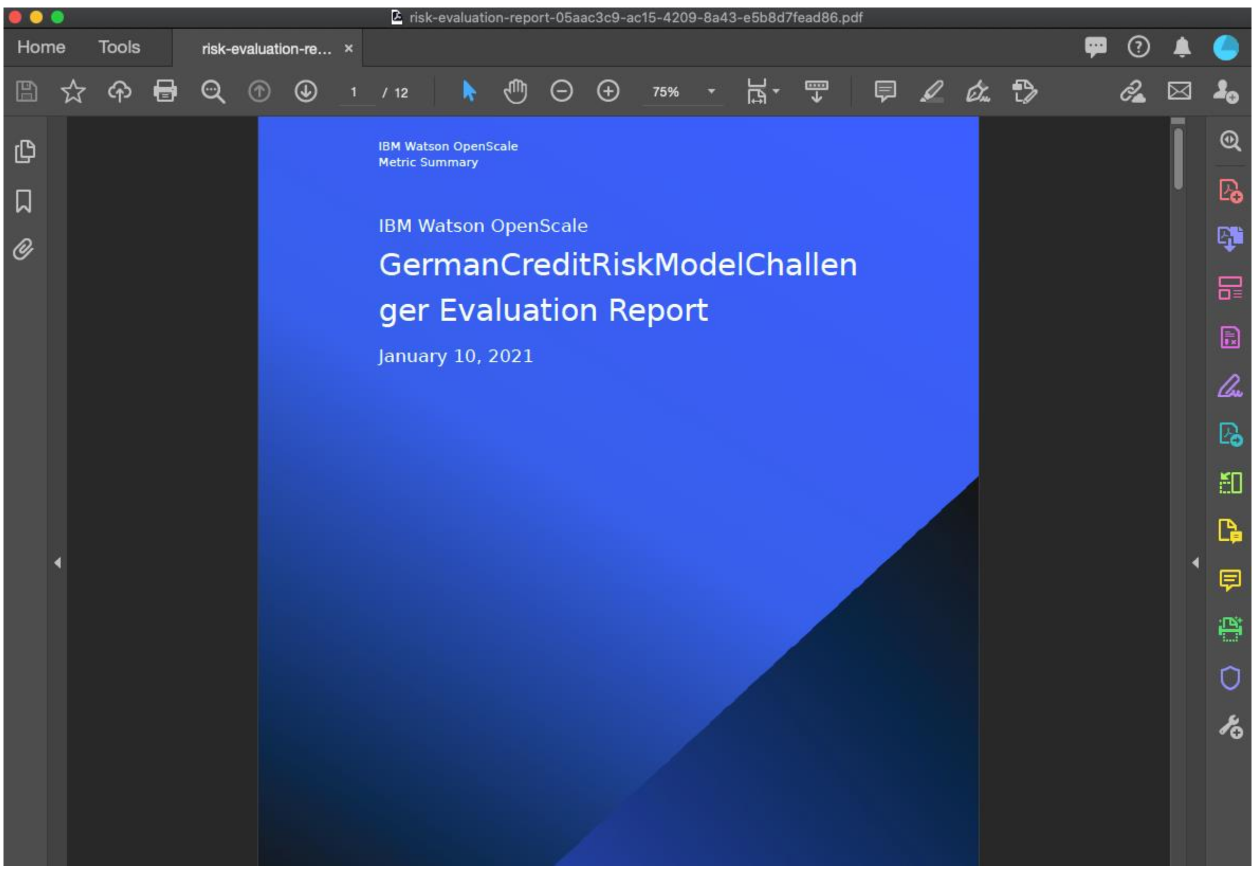
4. Examine the comparative results of both models and click the **X** to close the panel.



5. A final step is to deliver everything we have generated in a single report. Click on **Actions** and select **Download report PDF**. This may take a few minutes to prepare.



6. Browse the generated report.



Congratulations! You have completed the lab!!!

- ✓ Provisioned Watson OpenScale
- ✓ Viewed Fairness and Quality Metrics
- ✓ Viewed Drift Metrics.
- ✓ Explained a Transaction.
- ✓ Compared Pre-production Models and Generated a Report.