CENTENNIAL COLLEGE

BA 723 – Business Analytics Capstone

Project: Brewery Recipe Optimization – Monitoring and Governance

- Max Sergio Causso Fretel - 301455365

CENTENNIAL COLLEGE

1. Model Risk Overview

Definition of Model Risk: In the context of our Alcohol by Volume (ABV) Prediction Platform, model risk refers to the possibility of adverse outcomes (financial loss, compliance violations, or decision errors) arising from the model's inaccuracies or misuses. Model risk is present whenever an insufficiently accurate or mis-specified model is used to make decisions (Investopedia, 2020). In our use case, the model predicts the ABV of beer based on brewing parameters. If the model's prediction is wrong (for example, deviating significantly from the true ABV), it could lead to incorrect labeling or misguided business decisions. Since ABV is a regulated attribute that must be printed on labels for beverages ≥1.1% alcohol (Canadian Food Inspection Agency, 2020), a prediction error beyond allowed tolerance can have concrete negative impacts on the business.

Sources of Risk: Several sources of model risk have been identified for the ABV predictor, including:

- **Model Complexity (Glass-box vs Black-box):** I deployed two modeling approaches – a Ridge Regression (a "glass-box" linear model) and a Random Forest (a "black-box" ensemble model). The Random Forest's complexity and lack of interpretability introduce higher risk: it may fit spurious patterns or interactions that are not truly causal, and it's harder to explain its predictions. This opacity means if it does err (e.g. predicting an implausible ABV for a given input), it may be difficult to diagnose why. In contrast, the Ridge Regression is simple and transparent (essentially capturing the known linear relationship ABV ≈ (OG – FG) * 131 (ABVCalculator.ca, n.d.)), so its behavior is more predictable. Relying on a black-box for a critical output like ABV could thus elevate model risk due to complexity.

- **Data Limitations and Bias:** The model was trained on historical brew data which may not encompass all possible beer styles or brewing conditions. For instance, if the training set included mostly moderate-strength ales, the model might be less accurate for extreme cases (e.g. very high gravity stouts or unusual fermentations). Limited representation can lead to distributional shift risk – the model may encounter inputs outside the range or pattern of the training data. As a concrete example, Original Gravity (OG) in our training data ranged roughly from 1.030 to 1.090 SG; if a new recipe has OG 1.120 (very high sugar content), the model might extrapolate unpredictably. Similarly, other features like bitterness (IBU) or mash efficiency might shift if the brewery's process changes (e.g. improved equipment could raise the typical OG or efficiency). These data limitations mean the model could become less reliable as the business introduces new recipes or techniques, underscoring the need for ongoing monitoring.

- **Business Impact:** The predictions from this model directly influence product labeling and possibly operational decisions. A significant model error in ABV could have financial and reputational consequences. From a regulatory perspective, alcoholic beverages must have accurate ABV on their labels. In the US, for example, malt beverage labels are allowed only a ±0.3% ABV deviation from the stated value (Alcohol and Tobacco Tax and Trade Bureau, 2023). If our model underestimates a beer's strength by,

say, 0.5% ABV, the product would be non-compliant with labeling laws. Regulatory bodies like the CFIA and provincial liquor authorities (e.g. the LCBO in Ontario) enforce such rules by testing products. In fact, every product sold through the LCBO undergoes laboratory testing for alcohol content and label compliance (Liquor Control Board of Ontario, n.d.). A model that consistently mis-predicts ABV could trigger recalls, re-labeling, or fines. From a consumer safety standpoint, an understated ABV might lead a consumer to unknowingly consume more alcohol than intended, whereas an overstated ABV could unfairly alarm or dissuade customers. Thus, the business impact of model errors is non-trivial: it spans regulatory sanctions, loss of customer trust, and potential health/liability issues.

- **Regulatory and Compliance Considerations**: While predicting ABV is not as high-stakes as, say, credit underwriting or medical diagnosis, it does intersect with regulatory compliance. Alcohol labels are regulated; inaccuracies can be viewed as misrepresentation. The ABV Prediction Platform must therefore be governed under a framework that ensures accuracy, transparency, and accountability. Regulations don't explicitly forbid using an ML model for ABV, but any such model's output must fall within legal tolerances. Our governance approach treats the model as a decision-making tool that must be validated and monitored just like any other method of determining label ABV (such as lab measurements). I incorporate checks to mitigate this risk – for example, comparing the model's output with the standard ABV calculation formula as a sanity check, since by definition ABV ≈ (OG – FG) × 131.25 for beer (ABVCalculator.ca, n.d.). If the model's output ever deviates substantially from the well-established formula beyond normal error bounds, that's a red flag of model failure. In summary, the model risk here arises from potential inaccuracy, data scope, and complexity – amplified by the fact that the predictions carry business and compliance weight. Effective governance, as detailed in the following sections, aims to mitigate these risks through careful validation, monitoring, and controls (Datatron, 2021).

2. Initial Model Fit and Evaluation

Performance on Holdout Data: After training the models on the historical brew dataset (build data), I evaluated both the Ridge Regression and the Random Forest on a holdout test set (data not seen during training). This initial evaluation provides a baseline of how well each model predicts ABV on new data. The Ridge Regression (linear model) achieved an R^2 of approximately 0.95 on the test set, with a Mean Absolute Error (MAE) of about 0.20 % ABV. In comparison, the Random Forest model achieved a slightly higher R^2 of ~0.98 and an MAE of around 0.12 % ABV on the same test data. In practical terms, the Random Forest's predictions were a bit more accurate on average – for example, it might predict a beer's ABV within ±0.12% of the true value on average, whereas the Ridge was off by ±0.20%. Both models had high R^2 scores, indicating that the majority of variance in ABV was explained by the input features. This is not surprising, given that ABV is fundamentally determined by OG and FG through a near-linear relationship. The strong performance of the linear model confirms that our data is consistent with brewing science (i.e., the roughly linear formula), while the Random Forest's even higher accuracy suggests it may be capturing small nonlinear nuances or interactions in the data.

Glass-box vs Black-box Comparison: While the Random Forest had a performance edge, it is important to weigh accuracy against interpretability and robustness. The Ridge Regression is essentially a transparent mathematical equation. It found, as expected, that the difference (OG – FG) is the dominant predictor of ABV (with a coefficient ~131, in line with the standard formula (ABVCalculator.ca, n.d.)). This glass-box model provides clear insight: for each 0.001 increase in (OG–FG), ABV goes up by roughly 0.13%. I can explain its predictions easily to stakeholders (e.g., "Beer X had OG 1.060 and FG 1.010, so the predicted ABV is ~6.6%"). In contrast, the Random Forest leverages many decision trees and can consider interactions (perhaps it noticed patterns like very high IBUs correlating with slightly higher attenuation in our dataset). However, its reasoning is not easily interpretable – it's a black box in which I cannot straightforwardly trace how each input contributes to the output. This means that while the Random Forest might be slightly more accurate, it comes at the cost of transparency. In a high-compliance environment, that trade-off might be unacceptable; for our scenario, I mitigate this by using the interpretable model (Ridge) as a benchmark and sanity check for the black-box model's predictions.

3. Model Drift Monitoring

Once the model is in production, a key risk is model drift – where the model's performance degrades over time due to changes in the data or environment. I have established a monitoring framework to detect such drift early, using statistical tests like Population Stability Index (PSI) for continuous variables and Chi-square tests for categorical variables. Model drift can manifest in two ways: data drift (the input data distribution shifts from what the model saw in training) or concept drift (the underlying relationship between inputs and output changes, e.g. due to process changes). Our monitoring focuses primarily on data drift as an early indicator, since significant data distribution changes often precede or accompany concept drift (Datatron, 2021).

PSI for Continuous Features: Population Stability Index (PSI) is a widely used metric to quantify distribution change. I calculate PSI for key numeric features by binning the training data distribution and comparing it to the same bins for recent production data. A PSI below 0.1 is generally considered stable (no drift) – the model is operating within its expected domain. If any feature's $0.1 \leq PSI < 0.2$, I classify it as moderate drift. The model can still be used, but I increase the monitoring frequency and begin investigating the cause of the shift. For example, if PSI for OG = 0.15, that suggests a mild but notable change – perhaps the brewery is brewing slightly higher-gravity beers on average. If $PSI \geq 0.2$ for any feature or for the model's overall prediction output, it triggers a drift alert. This indicates a significant population change – the model is likely no longer reliable without intervention. In this case, our procedure is to schedule a model review and likely retraining with newer data (Katonic AI, 2022). For instance, if the mash efficiency's PSI jumps to 0.25, I suspect a substantial process change (maybe new equipment improved efficiency) and would plan to retrain the model to adapt.

Chi-Square for Categorical Shifts: Our current model features are mostly continuous, but if we consider any categorical factors in the future (for example, if a later model version includes a categorical input like yeast strain or beer style), I would use a Chi-square test to monitor drift in those distributions. A Chi-square test compares the frequency distribution of categories in current production data to that of the training data to see if they are significantly different. For instance, if the training data was 50% ales and 50% lagers, but in the last month the brewery

produced 90% ales and only 10% lagers, the Chi-square test would flag this as a significant change (p-value below threshold). Such a shift might matter if the model has different accuracy for different categories. In our present model, an analogous approach can be taken by bucketing a continuous feature into categories (e.g., low/medium/high OG) and applying a Chi-square test – though in practice PSI/CSI give essentially the same insight for numeric features. The important point is that I have statistical monitoring at both the feature level and overall model level to catch drift.

Example: Chi-Square Test for OG Drift: As an illustrative calculation, I performed a Chi-square test to compare the distribution of Original Gravity (OG) between the training data and a simulated set of production data. I binned OG into four ranges (low to very high OG) and tabulated the frequencies:

OG Range (specific gravity)	Training % (Expected)	Production % (Observed)	Training (Expected Count)	Production (Observed Count)
Low (≤ 1.040)	10%	5%	10	5
Medium (1.041 – 1.055)	40%	25%	40	25
High (1.056 – 1.070)	40%	50%	40	50
Very High (> 1.070)	10%	20%	10	20
Total	100%	100%	100	100

Here I assumed 100 observations in each dataset for simplicity. The "Training %" column represents the baseline distribution of OG (from training), and "Production %" is the new data distribution. The expected counts for the production set are calculated by applying the training percentages to the production sample size (e.g., expected 10% of 100 in Low OG = 10). The Chi-square statistic is then computed by summing (Observed – Expected)² / Expected for each category:

- For Low OG: $(5 – 10)^2 / 10 = 25/10 = 2.5$
- For Medium OG: $(25 – 40)^2 / 40 = 225/40 = 5.625$
- For High OG: $(50 – 40)^2 / 40 = 100/40 = 2.5$
- For Very High OG: $(20 – 10)^2 / 10 = 100/10 = 10$

Summing these yields a Chi-square statistic ≈ 20.62. The degrees of freedom (gl) in this test are $k – 1 = 4 – 1 = 3$, since there are 4 OG categories. At a significance level of 0.05, the critical Chi-square value for 3 degrees of freedom is about 7.815. Our calculated value (20.62) far exceeds this, meaning we reject the null hypothesis that the OG distribution is unchanged – there is a statistically significant drift in OG. In practice, this large shift would immediately prompt an investigation and likely retraining of the model to account for the new OG profile. The example illustrates how a Chi-square test can quantitatively confirm drift detected in a feature's distribution.

Degrees of Freedom: In the context of Chi-square tests, degrees of freedom (df) correspond to the number of independent categories that can vary without violating constraints. For a goodness-of-fit test comparing distribution across k categories, df = k – 1 (because the expected frequencies must sum to the total, leaving k–1 free values). In a contingency table (Chi-square test of independence), df = (rows – 1) * (columns – 1). In the OG drift example above with k = 4 categories, df = 3. The degrees of freedom are used to determine the Chi-square distribution's critical values for significance testing. Essentially, higher degrees of freedom shift the Chi-square threshold for what counts as "significant" drift at a given confidence level. We always report the df alongside the Chi-square statistic, as it contextualizes the result (e.g., "$\chi^2 = 20.62$, df = 3").

Drift Response Plan: Our governance policy clearly defines actions based on these monitoring metrics. Depending on the severity of detected drift (measured by PSI, Chi-square, or error rates), I take tiered actions to mitigate risk:

- **No action (Green):** If drift indicators remain below the defined threshold (e.g., PSI < 0.1 and no statistically significant change detected), the model is deemed stable. I continue regular monitoring but do not intervene.
- **Alert & Report (Yellow):** If mild drift is detected (for instance, a PSI in the 0.1–0.2 range or a Chi-square test barely significant), I raise an alert to the data science team and relevant stakeholders. The change is documented in a report, and I increase scrutiny – for example, performing a targeted evaluation of recent predictions vs. actuals to check if error is creeping up. I also monitor subsequent batches closely to see if the drift persists or was a temporary fluctuation.
- **Refit Model (Orange):** If moderate drift is detected that is likely to affect model performance (e.g., PSI ≥ 0.2 or model error on new data exceeds business thresholds, such as MAE above 0.3% ABV), I initiate a model retraining process. This involves collecting a fresh batch of labeled data (actual OG, FG, ABV outcomes from recent production) and refitting the model so it can learn the new patterns. In tandem, I investigate the root cause of the drift. For example, if IBU distribution shifted significantly, was there a change in recipe trends? If efficiency shifted, was new equipment installed? Understanding the cause helps determine if any one-off fixes are needed (such as adjusting a feature or adding a new input to the model) in addition to retraining. By retraining (and possibly recalibrating) the model, I ensure it stays accurate under the new data conditions.
- **Rebuild Model (Red):** If severe drift or concept drift is detected – meaning the data or relationships have changed so much that the current model may no longer be appropriate – I consider a full model rebuild. This is a higher-effort response: I would revisit the feature set and modeling approach and potentially develop a new model from scratch that better fits the new regime. This action is reserved for situations where simply refitting the existing model is insufficient (for example, if an entirely new range of inputs is being seen, or the brewing process changes fundamentally such that the original model's form is no longer valid). Rebuilding also involves revalidating and documenting the new model thoroughly before deployment.

These tiered responses align with a risk-based approach: minor drifts result in heightened monitoring, whereas major drifts trigger corrective action (retrain or rebuild) to maintain model

integrity. This ensures we neither under-react to meaningful changes nor over-react to negligible noise. By continuously tracking these metrics and following the response plan, I ensure the model does not quietly "age" and produce biased outputs. This ongoing monitoring and escalation framework acts as an early warning system so I can react before significant mis-predictions occur, in line with best practices in model risk management.

4. Risk Tiering Classification

I classify the ABV prediction model's overall model risk level using a framework inspired by the European Commission's risk-tiering approach for AI, adapted to our business context. The EU's draft AI Act defines certain high-risk AI applications (e.g. those affecting critical infrastructure, safety, or fundamental rights) (ModelOp, 2023), and by contrast deems other uses as limited or low risk when they are narrow and have minimal impact. Our ABV model is a narrow regression model used in a brewing context – far from the life-critical or rights-impacting systems that regulators would label as "high-risk." However, it does have compliance implications and business impact, so I consider it more than trivial. I define the tiers for our model as follows:

- **Low Risk:** An AI/ML model whose errors or misuse would have negligible impact on business operations, consumers, or compliance. These are typically models used for internal insights or minor optimizations, where any failure causes at most minor inconvenience. In our context, if the ABV model were used only as an optional decision support tool (for instance, giving brewers a quick estimate before they do a proper lab test) and all outputs were double-checked by humans, the risk could be considered low. In that scenario, the model is not solely responsible for any decision; human expertise and lab measurements ultimately ensure accuracy. Low-risk also assumes that even if the model is wrong, the consequence (e.g. a slightly off internal estimate) does not translate directly to a consumer-facing error or regulatory breach. (Under the EU AI Act analogy, a low-risk system might still require transparency to users that they're dealing with an AI (ModelOp, 2023), but not stringent oversight.)

- **Moderate Risk:** This tier involves models that have a direct impact on business outcomes or customers but in scenarios that are not life-critical and that have mitigation controls. I classify our ABV Prediction Platform as Moderate Risk. The model's predictions are used to print ABV on product labels (a regulatory requirement) and to ensure tax/quality compliance. Mistakes in this area can lead to moderate consequences – for example, a beer with mislabeled ABV might result in a product recall or penalty, and could slightly affect consumer experience or safety (a stronger beer than labeled might cause someone to consume more alcohol than they realized). These are important issues, but they are bounded in impact (affecting finances and compliance rather than someone's life or rights). Moreover, we have mitigation measures: periodic lab tests by Quality Assurance can catch large deviations, and the brewing team reviews the model outputs. The model is also interpretable (especially if using the Ridge regression version), which adds a layer of control – I understand how it works and can detect if it starts behaving oddly. In EU terms, this is akin to a "limited risk" AI performing a narrow task with human oversight (ModelOp, 2023). I impose transparency and monitoring requirements appropriate for this risk level, but it doesn't trigger the heavy regulatory scrutiny of high-risk AI. The moderate classification does, however, prompt us to implement solid

governance (documentation, monitoring, version control – see Section 6) to manage the risk.

- **High Risk:** Models that, if incorrect or abused, could lead to severe harm – e.g. endangering human safety, violating rights, or causing major financial loss – fall in this category. Examples would be an AI controlling critical infrastructure, making medical diagnoses, or driving a car (ModelOp, 2023). Our ABV model clearly does not fall in this tier. It does not have the ability to directly injure or discriminate against people, and its domain is confined to beverage alcohol content. Even in a worst-case scenario (e.g., the model consistently underestimates ABV and many products are mislabelled), the likely outcomes are product relabeling and some upset customers or regulators – serious for business, but not life-threatening. Therefore, I do not consider this a high-risk AI system. Consequently, we are not subjecting it to the stringent controls that a high-risk system would require (such as mandatory external audits or registration with authorities in some regulatory regimes). If in the future the model's scope expanded (imagine it being used to automatically regulate alcohol tax payments or control a safety-critical brewing process), I would reassess this classification.

Conclusion – Tier Assignment: The ABV Prediction Platform is classified as Moderate Risk (roughly corresponding to a "limited risk" AI system under EU guidance, but with some business-critical aspects). This means I treat it with appropriate rigor in terms of governance: I ensure clear documentation, human oversight, and rigorous monitoring. But I acknowledge it is not in the realm of high-risk AI – it is a narrow, well-understood application. This classification guides our governance strategy: for a moderate risk model, I implement controls to minimize the risk (through validation, monitoring, version control, etc.) but also keep the process efficient enough to not stifle the model's utility. Importantly, even as a moderate risk model, any use of its predictions must be transparent to stakeholders. For instance, if the ABV on a label was model-predicted rather than measured in a lab, our internal policy is to validate that prediction against a lab sample periodically to maintain confidence. Overall, aligning with the European-style risk-based approach helps ensure we neither under- nor over-regulate the model – we apply just enough governance to match the impact of potential failures (ModelOp, 2023).

5. Variable-Level Monitoring

In addition to overall model performance monitoring, I perform variable-level monitoring to ensure each input feature remains within valid ranges and retains a consistent distribution. At model build time, I captured baseline summary statistics for each key input variable; these serve as reference points for detecting drift in individual features and for enforcing sanity checks on incoming data.

Build Data Baselines: In the training dataset, the key brewing parameters had the following typical values: Original Gravity (OG) averaged around 1.055 specific gravity (with most values roughly 1.040–1.070, min ~1.030, max ~1.090), Final Gravity (FG) averaged ~1.012 (most 1.000–1.025), Bitterness (IBU) averaged ~35 (ranging from very low ~5 up to ~100), and Mash Efficiency around 75% (typical range ~50–90%). These ranges align with standard brewing practices – for example, most beers target an OG near 1.050 (Brewer's Friend, 2023). Values outside these ranges (e.g., OG > 1.100, FG > 1.030, IBU > 100, or efficiency > 95%) were not

present in training and would be considered out-of-domain if encountered in production. Such outliers indicate either a novel brewing scenario or a data error.

Validation Rules for Inputs: Based on the above, I established validation rules and guardrails for data fed into the model:

- For each numeric feature, I set plausible minimum/maximum bounds to catch unrealistic values. For example, OG is capped at 1.120 – if any brew reports OG higher than that, the system will log a warning and cap the value at 1.120 for the prediction. Similarly, FG is floored at 0.990 (since readings below 0.990 are not expected under normal fermentation), IBU is capped at 120, and efficiency is constrained between 0% and 100% (with warnings if outside the 50–90% typical band). These caps prevent the model from extrapolating into absurd ranges (which could produce unreliable outputs) and serve as data quality checks by flagging potential data entry or sensor errors.

- I define business rules for missing or null values. For critical features like OG and FG, our policy is that if either is missing for a batch, we should **not** rely on the model's prediction (since ABV calculation fundamentally depends on OG–FG). Instead, that batch would fall back to a direct lab measurement or use a standard fermentation formula. I do not impute OG or FG because their relationship to ABV is grounded in physical chemistry. For less critical features like IBU or efficiency, if a value is missing, I may impute a conservative default (e.g., assume IBU = 0 if unknown, or use average efficiency ~75%). However, missing data is rare in our pipeline because brew records are typically complete; our governance documentation notes: "If IBU or efficiency is missing, use the latest typical value or mark the record for review." In all cases, any imputation or substitution is logged for transparency.

- At scoring time, if any input falls outside its predefined valid range (after applying caps), the system raises an **alert**. For example, if OG comes in as 1.150 (which would be capped to 1.120 for the model), an alert is sent to the data science team and brew operations. This indicates either a data issue or a truly out-of-distribution brew (very high gravity) for which the model was not trained. In either case, human attention is warranted. This ties back to risk management – we don't blindly trust predictions on out-of-bound inputs without at least waving a red flag to prompt investigation.

Variable Drift Monitoring (CSI): To complement overall PSI monitoring, I keep an eye on each input's distribution over time using the Characteristic Stability Index (CSI), which is essentially the PSI applied to individual features (Katonic AI, 2022). By tracking CSI for OG, FG, IBU, and efficiency, I can pinpoint which specific variable is shifting if the model's performance degrades. For example, suppose over a year I notice the average OG creeping upwards (perhaps the brewery is making stronger beers due to market demand). I might see OG's CSI = 0.25 when comparing this year's data to last year's – a significant shift indicating the input distribution has changed. Even if the model is still performing "okay" in terms of error, this drift would prompt us to retrain the model to recentre it on the new OG range (and possibly re-evaluate if our linear assumption still holds at those extremes). Similarly, if plant-wide efficiency improved (say new equipment raises it from ~75% to ~85% consistently), the efficiency CSI might be high. (Efficiency itself doesn't directly feed the ABV formula if OG and FG are measured, but if our model used efficiency as a feature to estimate FG or attenuation, a change

there could indirectly affect predictions.) Monitoring at the variable level also helps diagnose overall PSI alerts: if the model's overall prediction PSI goes into the alert range, I can look at the CSIs to see which feature contributed most. For instance, a big shift in IBU distribution might not actually affect ABV predictions much (since IBU is only loosely related to ABV), whereas a shift in FG distribution (maybe beers are finishing drier on average) would directly impact ABV. Knowing which variable drifted guides our retraining and investigation efforts.

Example of Variable Drift Impact: To illustrate why this matters, consider a concrete scenario: suppose in the training data the average FG was 1.012, but over time the brewery starts using a new yeast that attenuates more sugar, bringing the average FG down to 1.005 in recent batches. This is a substantial change in the FG distribution (the CSI for FG could well exceed 0.2, indicating significant drift). What does this mean for the model? A lower FG for a given OG means higher ABV – these beers are drier and stronger than before. If our model wasn't retrained, it might systematically under-predict ABV for these new batches (because it learned a typical attenuation level of ~75%, and now attenuation is ~90%). This variable-level shift would be caught by our monitoring. I would then update the model with data from these new brews, effectively recalibrating it to the new FG patterns. This example underscores why monitoring each key input is vital: even if OG and other factors remain the same, a drift in FG (or any single variable) can upset the model's accuracy. By tracking the means, standard deviations, and distribution shapes of each variable, I maintain a tight grip on the model's domain. It's worth noting that our Power BI monitoring dashboard (discussed later) displays monthly trends for OG, FG, IBU, efficiency, etc., against their historical ranges. This visual tracking, combined with CSI metrics, ensures that any creeping changes in the brewing process are not missed. Our governance documentation also explicitly lists for each variable: the expected range, the rationale for those ranges (e.g., citing brewing literature or historical data), and the action plan if out-of-range values occur (such as halting model predictions for that batch, triggering a data review, etc.). All these practices keep the model's operation transparent and within the guardrails of its design assumptions.

6. Governance Recommendations

To ensure the long-term reliable and responsible use of the ABV Prediction Platform, I have implemented a set of governance measures. These address model versioning, documentation, access control, and alignment with our existing monitoring infrastructure (Power BI dashboards and Azure cloud services).

- **Model Versioning:** Each time the model is trained or updated, it is assigned a new version identifier and logged in a model registry. This registry (which could be the Azure Machine Learning model registry or a simple internal database) tracks the model's lineage – including training data used, training date, key performance metrics, and notes on what changed from prior versions. Versioning the model is critical so that I know exactly which model produced which ABV predictions. For example, if a question arises about a product's labeled ABV, I can trace it to the specific model version and parameters used at the time. Key metadata recorded for each version includes: features used (and their definitions), training dataset timeframe, algorithm and hyperparameters, performance metrics (train/validation/test R^2, MAE, etc.), and validation results (e.g., did it meet error tolerance on a holdout set?). This practice ensures traceability and aligns

with model risk management best practices (Datatron, 2021). If an issue is discovered, I can audit prior versions to see when it was introduced. In our Azure ML pipeline, I automate this by registering the model after training and tagging it with a version number and a brief description (e.g., "v2.0 – retrained with data through 2025, includes new yeast strain data"). Stakeholders (the brewing team, QA, etc.) are notified of new versions and given a summary of changes. This controlled approach avoids the risk of "model drift" from ad-hoc updates – any change is deliberate, reviewed, and recorded.

- **Documentation and Transparency:** I maintain thorough documentation for the model, effectively creating a *Model Card* or similar documentation file as required by the BA723 governance standards. This includes: (a) Model Description – the purpose of the model, how it works (e.g., "predicts ABV from OG, FG, etc. using regression"), and its intended scope of use (e.g., only for beer recipes under typical brewery conditions). (b) Assumptions & Limitations – documenting assumptions (e.g., the model assumes accurate OG/FG measurements and typical beer formulations) and known limitations (e.g., it may not be valid for wines or spirits, or for beers outside the range of the training data). I include the known valid ranges of inputs (from Section 5) and note what kinds of situations the model has not seen (e.g., non-beer fermentations). (c) Performance Evaluation – the initial training and holdout metrics (R^2, MAE) and any ongoing re-evaluation results, along with acceptable error thresholds (for example, I state: "The model is expected to predict ABV within ±0.3% for 95% of products, in line with regulatory tolerance (Alcohol and Tobacco Tax and Trade Bureau, 2023)"). (d) Testing and Validation – how the model was validated (including back-testing against lab results) and the ongoing monitoring procedures (PSI/CSI metrics as discussed above). This serves as evidence that the model performs as intended in the real world (Datatron, 2021). (e) Governance and Approval – which stakeholders or committees reviewed and approved the model for deployment (e.g., data science lead, head brewer, compliance officer), including dates and any conditions (such as "approved for use on beers up to 10% ABV; stronger products require additional validation"). (f) Version History – a log of changes for each version (as noted in the versioning process). (g) Access and Security – documented in a section on who is allowed to access or modify the model and data.

This documentation is stored in an accessible repository (e.g., a SharePoint site or as part of the Azure ML workspace) and is updated whenever the model is retrained or modified. It provides transparency so that any auditor or new team member can understand what the model is, how it operates, and what its boundaries are. In highly regulated industries like finance, such documentation is mandatory; while our use case is less regulated, I apply a similar rigor to ensure accountability and clarity.

- **Access Control:** I implement role-based access control for the model and its predictions. Only authorized personnel (e.g., data scientists, the brewing QA team, and specific IT/DevOps staff) have the ability to modify the model or the input data feeding it. Access control addresses questions like "Who has access to the model, and has there been any unauthorized change?" (Datatron, 2021). For instance, the model's code and training data are stored in a secure Git repository with limited write access. The production model endpoint (if deployed as a service) is secured by API keys or Azure Active Directory authentication, meaning only the brewing application and authorized users can call it. I

also log every prediction request and response (along with the model version) in an audit log. This way, if someone were to abuse or tamper with the model, we have a detailed audit trail. Furthermore, I maintain separate environments for development, testing, and production – models and code are promoted to production only after proper testing and approval, to avoid any unreviewed changes going live. I periodically review user access lists to ensure only current team members have permissions, following the principle of least privilege.

- **Alignment with Monitoring Infrastructure:** Our model monitoring is integrated into the existing analytics and cloud infrastructure. In Azure, I leverage Azure Machine Learning's dataset monitors and Application Insights for automated tracking. For example, Azure ML's Data Drift monitoring service can be configured to track features like OG, FG, etc., comparing incoming scoring data to the training baseline (Microsoft, n.d.). I have enabled data collection on the deployed model – inputs and outputs are captured to Azure Blob Storage in a structured format (Microsoft, n.d.). These collected data are the basis for our PSI/CSI calculations. I schedule an Azure ML pipeline (or use the built-in monitoring tools) to compute PSI monthly and output the results. We feed these monitored metrics into our Power BI dashboards for visualization. Microsoft's guidelines suggest using Power BI to analyze collected model data for drift and performance (Microsoft, 2025), and we have done exactly that. Our Power BI report (accessible to the brewmaster and data science teams) shows time-series of ABV prediction errors (once actual lab ABVs are available for comparison), histograms of feature distributions vs. training baseline, and current PSI/CSI values with a traffic-light coloring scheme (green = stable, yellow = moderate drift, red = needs retrain). By analyzing this dashboard, we make informed decisions about when to retrain or adjust the model (Microsoft, 2025). For example, the dashboard might highlight: "PSI for OG = 0.12 (yellow, moderate drift) this quarter," which would prompt discussion in the weekly operations meeting. We also configure automated alerts: if any PSI/CSI exceeds its threshold or if prediction error on new data exceeds a set limit, an email/Slack alert is sent to the responsible team. Additionally, Azure allows triggering pipelines based on drift monitor results (Microsoft, n.d.). I plan to integrate an automated retraining trigger: for instance, if PSI ≥ 0.2 for a key feature, it can cue a retraining job (subject to human review before deployment). While full automation is technically possible, our governance will likely require a human sign-off before any newly retrained model goes live, to verify that nothing odd happened during retraining. Finally, the Power BI dashboards ensure that not only data scientists but also domain experts (brewers, quality managers) have visibility into the model's status. I include brief explanations on the dashboard (e.g., definitions of PSI and the meaning of thresholds) so that non-data folks can interpret it. This cross-functional transparency aligns with the principle of human oversight in AI governance (ModelOp, 2023) – the brewing team can question the model if they see something off in the metrics. We have a governance protocol to conduct a Model Performance Review quarterly, where we use the Power BI report to check that the model is still meeting business needs. In these meetings, we discuss any emerging trends or issues (e.g., if drift is gradually increasing or if error rates are creeping up) and decide on actions (such as scheduling a retraining or investigating data collection processes). This continual review process ensures we proactively maintain model quality and compliance.

In summary, through a combination of proactive monitoring, clear documentation, controlled access, and periodic reviews, we maintain strong governance over the ABV Prediction Platform. This governance ensures that the model remains accurate, reliable, and compliant with regulatory and business expectations over time. By adhering to the risk-based framework and implementing the recommendations above, we can confidently use the model's predictions in production, knowing that we have the oversight and processes in place to address any issues that arise.

CENTENNIAL COLLEGE

References

BA723 documentation presentation governance M25 [Lecture slides]. (2025). Centennial College.

Investopedia. (2020). *Model risk: Definition, management, and examples.* Retrieved from https://www.investopedia.com/terms/m/modelrisk.asp

Canadian Food Inspection Agency. (2020). *Labelling requirements for alcoholic beverages – Alcohol by volume declaration.* Retrieved from http://inspection.canada.ca/en/food-labels/labelling/industry/alcoholic-beverages

ABVCalculator.ca. (n.d.). *ABV calculator for beer, wine, mead, cider, and cocktails.* Retrieved from https://www.abvcalculator.ca/

Alcohol and Tobacco Tax and Trade Bureau. (2023). *Malt beverage labeling: Alcohol content (27 CFR 7.65).* Retrieved from https://www.ttb.gov/regulated-commodities/beverage-alcohol/beer/labeling/malt-beverage-alcohol-content

Liquor Control Board of Ontario. (n.d.). *Product testing | Doing business with LCBO.* Retrieved from https://www.doingbusinesswithlcbo.com/content/dbwl/en/basepage/home/quality-assurance/quality-assurance-policies---guidelines/product-testing.html

Datatron. (2021). *What is model governance and why it's important.* Retrieved from https://datatron.com/model-governance/

Katonic AI. (2022). *Use drift detection to ensure that your shiny AI models don't lose their luster!* Retrieved from https://blog.katonic.ai/use-drift-detection-to-ensure-that-your-shiny-ai-models-dont-lose-their-luster/

Evidently AI. (n.d.). *Which test is the best? We compared 5 methods to detect data drift on large datasets.* Retrieved from https://www.evidentlyai.com/blog/data-drift-detection-large-datasets

ModelOp. (2023). *EU AI Act: Summary & compliance requirements.* Retrieved from https://www.modelop.com/ai-governance/ai-regulations-standards/eu-ai-act

Brewer's Friend. (2023). *Batch statistics for extract base recipes.* Retrieved from https://www.brewersfriend.com/extract-ogfg/

Microsoft. (n.d.). *Detect data drift on datasets (preview) – Azure Machine Learning.* Retrieved from https://learn.microsoft.com/en-us/azure/machine-learning/how-to-monitor-datasets

Microsoft. (n.d.). *Collect data on your production models – Azure Machine Learning.* Retrieved from https://learn.microsoft.com/en-us/azure/machine-learning/how-to-enable-data-collection

Microsoft. (2025). *Collect data from models in production – Azure Machine Learning.* Retrieved from https://learn.microsoft.com/en-us/azure/machine-learning/how-to-enable-data-collection

Microsoft. (n.d.). *Automating retraining in Azure ML CI/CD pipeline based on data drift (Q&A).* Retrieved from https://learn.microsoft.com/en-us/answers/questions/2168254/automating-retraining-in-azure-ml-ci-cd-pipeline-b