Final project sponsored by PPG Industries

- Fortune 500 Company
- Global supplier of paints, coatings, and specialty materials
- Largest coatings company in the world by Revenue
- Headquarters in Pittsburgh, PA and operates in 70+ countries around the world
- Please see ppg.com to learn more!

Data science, analytics, and machine learning, play an important role throughout PPG!!

For example, machine learning techniques are used to:

- Optimize logistics and supply chain
- Design products
- Improve quality
- Improve efficiency
- Analyze and improve customer experience

We will apply machine learning techniques to study a VERY important area within PPG

- **PPG has given us the opportunity to apply machine learning techniques to study their existing library of paint colors!!!!**

We will apply machine learning techniques to study a <u>**VERY**</u> important area within PPG

- **PPG has given us the opportunity to apply machine learning techniques to study their existing library of paint colors!!!!**

- <u>**IMPORTANT**</u>**:** You are working with REAL data provided by a REAL company!!!
  - **You <u>MUST</u> download the data from Canvas to your machines for the project.**
  - **Although the data have been approved for use in this project, you are <u>NOT</u> allowed to post the data online.**
  - **You are <u>NOT</u> allowed to share the data on Github, on a Blog, on a Website, or any Repository.**

- If you would like, you are allowed to discuss the project on your resume/CV and share your analysis approach with others.
  - You are NOT allowed to share the data.
  - You are allowed to share the models you have trained, how you preprocessed the data, and the modeling strategy.
  - You <u>**CANNOT**</u> share the input importance rankings, predictive trends, and final conclusions about difficult to predict paints!!!! This information is proprietary to PPG!!!!
  - Please contact Dr. Yurko if you are not sure what you are allowed to share outside the course.

We will apply machine learning techniques to study a <u>**VERY**</u> important area within PPG

- PPG currently provides numerous tools to help individuals and companies explore, plan, and design <u>**COLORS**</u> for their various applications.
  - <u>Here are a few examples</u> of the tools already available, if you're interested.
  - Some of the tools involve interacting with color specialists and designers.
  - Other tools give the user the ability to test out various colors in their rooms.
  - While other tools provide users with important information and global trends.

- **It's your job to see if machine learning techniques can learn patterns associated with top selling paint colors!**
  - Perhaps machine learning methods could recommend paints to users in the future…

Data Description

- You are provided with a TRAINING set consisting of **INPUTS** and **OUTPUTS**.
  - The TRAINING set is provided as a CSV file.
  - One row in the TRAINING set corresponds to 1 Paint color sold by PPG.

- The **INPUTS** consist of variables from two alternative color models.
  - RGB color model – variables `R`, `G`, and `B`
  - HSL color model – variables `Hue`, `Saturation`, and `Lightness`

- Two **OUTPUTS** are provided and each represents two different important aspects of the paint color.
  - `response` – a CONTINUOUS output associated with an important paint property
  - `outcome` – a BINARY output associated with the POPULARITY of the paint

Below is a snippet of the TRAINING set provided on Canvas, showing the variable names and a few values

R	G	B	Lightness	Saturation	Hue	response	outcome
172	58	62	dark	bright	4	12	1
26	88	151	dark	bright	31	10	1
172	94	58	dark	bright	8	16	1
28	87	152	dark	bright	32	10	0
170	66	58	dark	bright	5	11	0
175	89	65	dark	bright	6	16	0
90	78	136	dark	bright	34	10	0
194	106	53	dark	bright	10	19	0
171	68	107	dark	bright	1	14	0
122	151	59	dark	bright	21	25	0
0	121	88	dark	bright	24	14	0
88	140	58	dark	bright	22	19	0
144	82	132	dark	bright	36	14	0

You must train models for two different tasks

- **Regression task**: Train models to predict the important paint property, `response`, as a function of the color model **INPUTS**.
  - The important paint property is named `response` in the training set. Thus, the true name of the important paint property has been hidden from you.

- **Classification task**: Train models to classify if the paint is among the popular paint products sold by PPG based just on color model **INPUTS**.
  - The popularity is provided as BINARY variable, `outcome`, in the TRAINING set.
  - Popular paints are denoted as `outcome = 1` while less popular paints are denoted as `outcome = 0`.
  - The **EVENT** is thus `outcome = 1` and the **NON-EVENT** is `outcome = 0`.

Primary goals of the project are associated with learning which INPUTS are important!

- **Regression task**: Train models to predict the important paint property, `response`, as a function of the color model **INPUTS**.
  - Want to learn how the color model **INPUTS** influence the important paint property!
  - Are the inputs from one color model more influential on predicting the important paint property?

- **Classification task:** Train models to classify if the paint is among the popular paint products sold by PPG based just on color model **INPUTS**.
  - Want to learn how the color model **INPUTS** influence the popularity!
  - Are the inputs from one color model more influential on the probability the paint is popular?

Inputs and color models

- The **INPUTS** are associated with two alternative color models.
- The `R`, `G`, and `B` **INPUTS** are the RGB values from the RGB color model.
- The `Hue`, `Saturation`, and `Lightness` **INPUTS** are HSL associated values from the HSL color model.
- **IMPORTANT**: you are **NOT** required to read or learn about color theory or color models for this project!!!!!
  - The actual **INPUT** names are provided to give you context for the project.

Continuous output considerations

- The continuous output, `response`, can be between 0 and 100. The value of 100 is the upper bound, while the value of 0 is the lower bound.

- A value of 100 is not necessarily "better" than a value of 75, 50, or 25. The units are hidden from you, and "better" depends on the context that the paint is used for.

- What matters for your project is that…**bounded outputs are not appropriate for Gaussian likelihoods!**

- Thus, you will **NOT** model the `response` **OUTPUT** directly.

- Instead you will model the **LOGIT** transformed `response`!

Continuous output considerations

- Instead you will model the **LOGIT** transformed response!
- The **LOGIT** transformation converts `response` to an UNBOUNDED variable!!!
- The **LOGIT** transformation with an arbitrary lower and upper bound is applied as:

$$y = \mathrm{logit}\left(\frac{response - lower}{upper - lower}\right)$$

- The **LOGIT** transformation can be calculated via the `boot::logit()` function:

```
y=logit( (response - 0) / (100 - 0) )
```

Binary outcome considerations

- Pay close attention to the empirical proportion of the EVENT when interpreting the Accuracy and other classification performance metrics!!!!!

The project therefore consists of the following regression and classification tasks

Regression

- Predict the **LOGIT**-transformed `response`, `y`, as a function of the provided inputs: `R`, `G`, `B`, `Hue`, `Saturation`, and `Lightness`.

Classification

- Classify if the binary `outcome` is the EVENT as a function of the provided inputs: `R`, `G`, `B`, `Hue`, `Saturation`, and `Lightness`.

The project is open ended

- No template is provided.

- An Rmarkdown is provided to give an example of reading in the data.
  - It also shows how to calculate the **<u>LOGIT</u>**-transformed response, and setup the binary outcome for `caret`/`tidymodels`.
  - It also shows how to save a model object and load that model in again.

- Specific requirements are listed next, and those requirements can help guide you through the predictive modeling application.

Project consists of 4 main areas

- **Part i: Exploration**
  - It is always important to explore and study your data before starting any modeling exercise.
- **Part ii: Regression**
  - Fit non-Bayesian and Bayesian linear models.
  - Train, tune, and assess performance of simple and complex models with resampling.
- **Part iii: Classification**
  - Fit non-Bayesian and Bayesian generalized linear models.
  - Train, tune, and assess performance of simple and complex models with resampling.
- **Part iv: Interpretation**
  - Identify the best models, most important features, and the hardest to predict paints for the regression and classification tasks.

Part i: Exploration

- Visualize the distributions of variables in the data set.
  - Counts for categorical variables.
  - Histograms or Density plots for continuous variables. Are the distributions Gaussian like?
- Condition (group) the continuous variables based on the categorical variables.
  - Are there differences in continuous variable distributions and continuous variable summary statistics based on categorical variable values?
  - Are there differences in continuous variable distributions and continuous variable summary statistics based on the binary `outcome`?
- Visualize the relationships between the continuous inputs, are they correlated?
- Visualize the relationships between the continuous outputs (`response` and the **LOGIT**-transformed `response`, `y`) with respect to the continuous **INPUTS**.
  - Can you identify any clear trends? Do the trends depend on the categorical **INPUTS**?
- How can you visualize the behavior of the binary `outcome` with respect to the continuous inputs? How can you visualize the behavior of the binary `outcome` with respect to the categorical **INPUTS**?

Part ii: Regression - iiA) Linear models

Before using more advanced methods, you need to develop a baseline understanding for the behavior of the **LOGIT**-transformed response as a function of the inputs using linear modeling techniques.

Use `lm()` to fit linear models. You must use the following:

- Intercept-only model – no INPUTS!
- Categorical variables only – linear additive
- Continuous variables only – linear additive
- All categorical and continuous variables – linear additive
- Interaction of the categorical inputs with all continuous inputs main effects
- Add categorical inputs to all main effect and all pairwise interactions of continuous inputs
- Interaction of the categorical inputs with all main effect and all pairwise interactions of continuous inputs
- 3 models with basis functions of your choice
  - Try non-linear basis functions based on your EDA.
  - Can consider interactions of basis functions with other basis functions!
  - Can consider interactions of basis functions with the categorical inputs!

Part ii: Regression – iiA) Linear models

- You must therefore train 10 different models!

- **Which of the 10 models is the best?**
  - **What performance metric did you use to make your selection?**

- Visualize the coefficient summaries for your top 3 models.

- **How do the coefficient summaries compare between the top 3 models?**

- **Which inputs seem important?**

Part ii: Regression– iiB) Bayesian Linear models

- You have explored the relationships; next you must consider the UNCERTAINTY on the residual error through Bayesian modeling techniques!
- Fit 2 Bayesian linear models – one must be the best model from iiA) and the second must be another model you fit in iiA).
  - State why you chose the second model.
- You may use the Laplace Approximation approach we used in lecture and the homework assignments.
- Alternatively, you may use `rstanarm`'s `stan_lm()` or `stan_glm()` function to fit full Bayesian linear models with syntax like R's `lm()`.
  - Resources to help with `rstanarm` if you're interested:
    - How to Use the rstanarm Package (r-project.org)
    - Estimating Regularized Linear Models with rstanarm (r-project.org)
    - Extra examples also provided on Canvas.

Part ii: Regression– iiB) Bayesian Linear models

- After fitting the 2 models, you must identify the best model.
  - Which performance metric did you use to make your selection?

- Visualize the regression coefficient posterior summary statistics for your best model.

- **For your best model:** Study the posterior **UNCERTAINTY** on the likelihood noise (residual error), σ.
  - How does the `lm()` maximum likelihood estimate (MLE) on σ relate to the posterior **UNCERTAINTY** on σ?
  - Do you feel the posterior is precise or are we quite uncertain about σ?

Part ii: Regression – iiC) Linear models Predictions

- You must make predictions with your 2 selected linear models in order to visualize the trends of the **LOGIT**-transformed `response` with respect to the inputs.

- You may use non-Bayesian or Bayesian models for the predictions.

- You must visualize your predictive trends using the following style:
  - The primary input should be used as the `x`-aesthetic in a graphic.
  - The secondary input should be used as a facet variable – it is recommended to use 4 to 6 unique values if your secondary input is a continuous variable.
  - You must decide the reference values to use for the remaining inputs.

- Whether you use non-Bayesian or Bayesian models, **you MUST include the predictive mean trend, the confidence interval on the mean, and the prediction interval** on the (**LOGIT**-transformed) `response`.

- **You MUST state if the predictive trends are consistent between the 2 selected linear models.**

Part ii: Regression – iiD) Train/tune with resampling

<u>You must train, assess, tune, and compare more complex methods via resampling.</u>

- You may use either `caret` or `tidymodels` to handle the preprocessing, training, testing, and evaluation.

<u>You must train and tune the following models:</u>

- Linear models:
  - All categorical and continuous inputs - linear additive features
  - Add categorical inputs to all main effect and all pairwise interactions of continuous inputs
  - The 2 models selected from iiA) (if they are not one of the two above)
- Regularized regression with Elastic net
  - Add categorical inputs to all main effect and all pairwise interactions of continuous inputs
  - The more complex of the 2 models selected from iiA)
- Neural network
- Random forest
- Gradient boosted tree
- 2 methods of your choice that we did not explicitly discuss in lecture

You must use ALL categorical and continuous inputs with the non-linear methods

Part ii: Regression – iiD) Train/tune with resampling

- **You must decide the resampling scheme.**
  - That resampling scheme must be applied to ALL models!

- **Different models have different preprocessing requirements.**
  - You must decide the appropriate preprocessing options you should consider.

- **You must identify the performance metrics you will focus on to compare the models.**
  - You must identify the best model.

Part iii: Classification - iiiA) GLM

Before using advanced methods, you need to develop a baseline understanding of the event probability as a function of the **INPUTS** using generalized linear modeling techniques.

Use `glm()` to fit generalized linear models. You must use the following:

- Intercept-only model – no INPUTS!
- Categorical variables only – linear additive
- Continuous variables only – linear additive
- All categorical and continuous variables – linear additive
- Interaction of the categorical inputs with all continuous inputs main effects
- Add categorical inputs to all main effect and all pairwise interactions of continuous inputs
- Interaction of the categorical inputs with all main effect and all pairwise interactions of continuous inputs
- 3 models with basis functions of your choice
    - Try non-linear basis functions based on your EDA.
    - Can consider interactions of basis functions with other basis functions!
    - Can consider interactions of basis functions with the categorical inputs!

Part iii: Classification – iiiA) GLM

- You must therefore train 10 different models!
- **These models are consistent with the regression portion.**
  - **Did you experience any issues or warnings while fitting the generalized linear models?**
- **Which of the 10 models is the best?**
  - **What performance metric did you use to make your selection?**
- Visualize the coefficient summaries for your top 3 models.
- How do the coefficient summaries compare between the top 3?
- **Which inputs seem important?**

Part iii: Classification – iiiB) Bayesian GLM

- Next, you need to consider uncertainty via Bayesian methods!
- Fit 2 Bayesian generalized linear models – one must be the best model from iiiA) and the second must be another model you fit in iiiA).
  - State why you chose the second model.
- You may use the Laplace Approximation approach we used in lecture and the homework assignments.
  - Alternatively, you may use `rstanarm`'s `stan_glm()` function to fit full Bayesian linear models with syntax like R's `glm()`.
- After fitting the 2 models, you must identify the best model.
  - Which performance metric did you use to make your selection?
- Visualize the regression coefficient posterior summary statistics for your best model.

Part iii: Classification – iiiC) GLM Predictions

- You must make predictions with your 2 selected generalized linear models in order to visualize the trends of the event probability with respect to the inputs.

- You may use non-Bayesian or Bayesian models for the predictions.

- You must decide which inputs you wish to visualize the trends with respect to.

- You must visualize your predictive trends using the following style:
  - The primary input should be used as the `x`-aesthetic in a graphic.
  - The secondary input should be used as a facet variable – it is recommended to use 4 to 6 unique values if your secondary input is a continuous variable.
  - You must decide the reference values to use for the remaining inputs.

- **You MUST include the predicted mean event probability and the confidence interval whether you use non-Bayesian or Bayesian models.**

- **You MUST state if the predictive trends are consistent between the 2 selected generalized linear models.**

Part iii: Classification – iiiD) Train/tune with resampling

You must train, assess, tune, and compare more complex methods via resampling.

- You may use either `caret` or `tidymodels` to handle the preprocessing, training, testing, and evaluation.

You must train and tune the following models:

- Generalized linear models:
  - All categorical and continuous inputs - linear additive features
  - Add categorical inputs to all main effect and all pairwise interactions of continuous inputs
  - The 2 models selected from iiiA) (if they are not one of the two above)
- Regularized regression with Elastic net
  - Add categorical inputs to all main effect and all pairwise interactions of continuous inputs
  - The more complex of the 2 models selected from iiiA)
- Neural network
- Random forest
- Gradient boosted tree
- 2 methods of your choice that we did not explicitly discuss in lecture

You must use ALL categorical and continuous inputs with the non-linear methods

Part iii: Classification – iiiD) Train/tune with resampling

- **You must decide the resampling scheme.**
  - That resampling scheme must be applied to ALL models!

- **Different models have different preprocessing requirements.**
  - You must decide the appropriate preprocessing options you should consider.

- **You must identify the performance metrics you will focus on to compare the models.**
  - You must identify the best model.

- **Which model is the best if you are interested in maximizing Accuracy compared to maximizing the Area Under the ROC Curve (ROC AUC)?**

Part iv: Interpretation – ivA) Input Importance

- With the model training completed, you can now answer meaningful questions associated with the data!

- You must identify the best regression model and the best classification model.

- Identify the most important variables associated with your best performing models.

- Are the most important variables similar for the regression and classification tasks?
  - Does one of the color model **INPUTS** “dominate” the other variables?
  - Does one of the color model **INPUTS** appear to be not helpful at all?

- Based on your modeling results, do you feel the color model **INPUTS** alone help identify **POPULAR** paints????

Part iv: Interpretation – ivB) Input insights

- You must drill down further to gain additional insights into the patterns of the data!

- **You must identify the combinations of `Lightness` and `Saturation`:**
  - **That appear to be the HARDEST to predict in the regression and classification tasks**
  - **That appear to be the EASIEST to predict in the regression and classification tasks**

- Base your conclusions on the best performing regression and classification models.

- You should base your conclusions on the resampled HOLD-OUT sets and **NOT** on the TRAINING set!
  - Thus, save your resampled hold-out set predictions!

Part iv: Interpretation – ivC) Prediction insights

- You must visualize the trends associated with the HARDEST and EASIEST to predict `Lightness` and `Saturation` combinations with respect to the TWO most important continuous inputs.
  - Predictions should be made using the best performing models.

- You must visualize your predictive trends as a SURFACE plot using the following style:
  - The primary continuous input should be used as the `x`-aesthetic in a graphic.
  - The secondary continuous input should be used the `y`-aesthetic in the graphic.
  - You must use 101 unique values for both the `x` and `y` aesthetics.
  - You must use `geom_raster()` to create the surface plot.
  - The fill aesthetic of `geom_raster()` must be set to the **<u>LOGIT</u>**-transformed `response` for the regression predictions and the EVENT probability for the classification predictions.

- You must make the surface plot for the hardest to predict `Lightness` and `Saturation` combinations and again for the easiest to predict `Lightness` and `Saturation` combinations .
  - You must decide the reference values to use for the other inputs.

- Thus you must make 2 surface plots for the best performing regression model and 2 surface plots for the best performing classification model.

Part iv: Interpretation – ivC) Prediction insights

- What conclusions can draw from your surface plots?

- Are the trends associated with the HARDEST to predict combinations different from the trends associated with the EASIEST to prediction combinations?

Two additional methods

- You may use the same two methods for both the regression and classification portions of the project.
  - If however, you select a method that cannot be used for both regression and classification, then you will need to select an additional method.

- Potential methods to consider:
  - Support Vector Machines (SVM) – classification and regression
  - Naïve Bayes – classification
  - Generalized Additive Models (GAM) – classification and regression
  - Multivariate Additive Regression Splines (MARS) – classification and regression
  - Partial Least Squares (PLS) – classification and regression
  - Deep neural network – classification and regression
  - K-nearest neighbors – classification and regression
  - Stacked models

- Please see Ch 6 in the caret documentation for a complete list of available methods in `caret`.
- Please see the tidymodels parsnip list of available models for models available in `tidymodels`.

Interpretation and visualization help

- Chapter 16 in the HOML provides useful discussion on interpretable machine learning.

- Provides code examples for visualizing model behavior and interpreting the graphics.

Homework assignments include examples working with `caret`

- You may use `caret` to perform all preprocessing, resampling, tuning, and evaluation for the project.

- However, you may use `tidymodels` instead of `caret`.

- `tidymodels` provides modeling aligned with the philosophy of the `tidyverse`, created by the developers of `caret`.

- If you are interested to learn `tidymodels`, please see the homepage, and try some of the “Get Started” tutorials.

Applied machine learning examples available on Canvas provide both `caret` and `tidymodels` examples

- Week 01 – Airfoil example problem
  - Example EDA, linear models, and regression models with `caret`

- Week 02 and Week 03 – examples
  - Regression application with `tidymodels` – Concrete data
  - Binary classification application with `tidymodels` – Ionosphere data

Test set predictions

- A test set of input values will be provided in April.
- You must predict the continuous response and the event probability using this test set.
- You will upload your predictions to a website. The website will provide performance metrics associated with your predictions.
- More to come on this later!

BONUS points – report to PPG

- Create a PowerPoint presentation highlighting the major results, findings, and conclusions from your work.

- **You may earn up to 10 BONUS points.**

- Background/motivation material is not needed for the presentation.

- Presentation is open ended, but some recommendations to include:
  - Interesting visualizations from the EDA portion of the project.
  - Visualizations comparing model performance and selection of the best model.
  - Visualizations showing variable importances and/or coefficient summaries.
  - Visualizations showing model predictions with text interpreting the trends.

BONUS points – synthetic data

- In lecture, we have discussed the importance and usefulness of synthetic (fake) data in a complete Bayesian model workflow.

- **You may earn up to 15 BONUS points if you create your own synthetic data and demonstrate the ability to recover the model parameters that generated the data.**

- You may consider the regression problem OR the classification problem, but you must use Bayesian modeling techniques.

- Regardless you must use the following to earn the maximum bonus:
  - You must consider 1 categorical variable with 4 levels (unique values)
  - You must consider 3 continuous variables
  - You must specify the true functional (basis) relationship between the linear predictor and the inputs. You must specify the true parameter values.
  - You must generate small, medium, and large sample size data sets.
  - You must fit your model, assuming the correct functional (basis) relationship for the small, medium, and large sample sizes.
  - How well are you able to recover the true parameter values given the three training sample sizes?

BONUS points – low frequency categories

- The TRAINING data was prepared for you so that certain challenging issues were removed.
- The data were prepared by removing categories that are VERY LOW frequency.
- Low frequency categories cause problems for resampling methods!

BONUS points – low frequency categories

- However, if you would like learn how to deal low frequency categories you may work with the **BONUS** data set supplied on Canvas.

- The **BONUS** data set is larger than the TRAINING set and consists of all categories for the categorical inputs.

- Some of the categories are VERY LOW frequency!

BONUS points – low frequency categories

- You only need to focus on the classification task with the **BONUS** data.
- **You may earn up to 10 BONUS points** if you fit 2 classification models which account for:
    - Low frequency categorical input classes via LUMPING
    - Near zero variance features.
- You **MUST** use `recipes` with `caret` or `tidymodels` directly to earn the **BONUS**.
- The **BONUS** data has an additional variable, `challenge_outcome`, but you do **NOT** need to consider `challenge_outcome` for the low frequency categories BONUS.

BONUS points – imbalanced data

- **IMPORTANT**: As you SHOULD see in your data exploration, the binary outcome is imbalanced.
- However, the binary outcome does not qualify as a “RARE” event since the class imbalance is **NOT** less than 15%.
- However, if you would like learn how to deal with imbalanced data, you may work with the **BONUS** data set provided in Canvas.

BONUS points – imbalanced data

- The **<u>BONUS</u>** data set includes an additional column compared to the TRAINING data set.

- The **<u>BONUS</u>** data set column `challenge_outcome` is a BINARY outcome where `challenge_outcome = 1` is the EVENT and `challenge_outcome = 0` is the NON-EVENT.

- The `challenge_outcome` BINARY variable is more difficult because the EVENT is RARE!!
  - The Paint popularity was defined another way…which makes this problem more difficult!

BONUS points – imbalanced data

- You only need to focus on the classification task with the **BONUS** data.

- **You may earn up to 15 BONUS points** if you fit 3 classification models which account for:
  - Low frequency categorical input classes via lumping
  - Near zero variance features
  - Output class imbalance via SUBSAMPLING methods

- You **MUST** use `recipes` with `caret` or `tidymodels` directly to earn the **BONUS**.

- Subsampling is a technique which artificially balances the output classes. This artificial sampling must be accounted for and should NEVER be done manually.
  - Please see Ch 11 from the caret documentation if you would like to learn about subsampling in `caret`.
  - Please see the tidymodels Learn page on subsampling for dealing with class imbalance in `tidymodels`.
  - Please see this Julia Silge blogpost for another `tidymodels` example with class imbalance.

BONUS points – imbalanced data

- The imbalanced data bonus shares some aspects with the low frequency category bonus.
- It is recommended to try the low frequency category bonus first before attempting the imbalanced data bonus.

Project Submission

- You must submit the RMarkdown source (.rmd) files and the associated rendered HTML documents.
- It is recommended that you create separate rmd files for different portions of the project. It is difficult to read and debug one enormous file!
- You must upload the separate HTML files. Do not zip files!
    - This is so that I can read the HTML in the Canvas grading system. Saves me a lot of time
- Project must be submitted no later than **Wednesday April 17th, 2024**