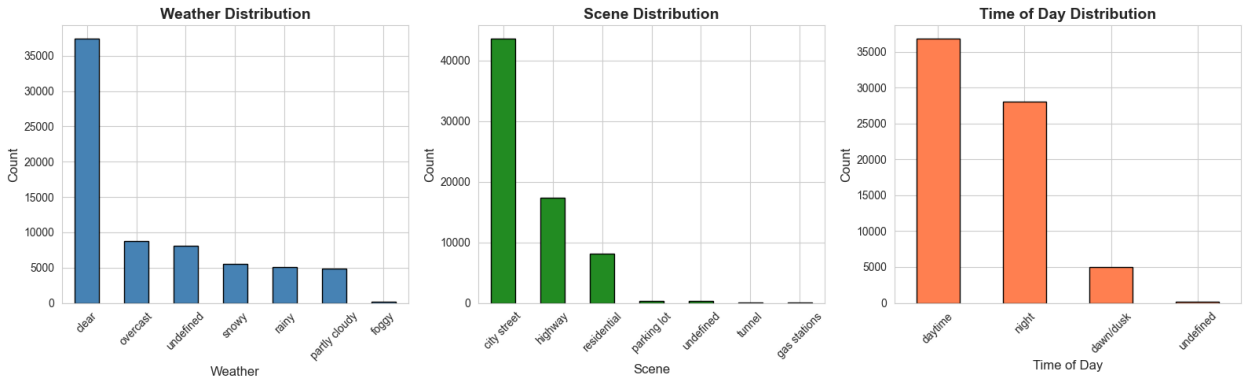
Object Detection on BDD100K using YOLOv11

Bbd100k dataset is huge 5.4 GB and over 70000 images in the train and 20000 in test and 10000 in validation set

So, I decided to do the EDA analysis on the whole dataset because we just need to process the JSON files

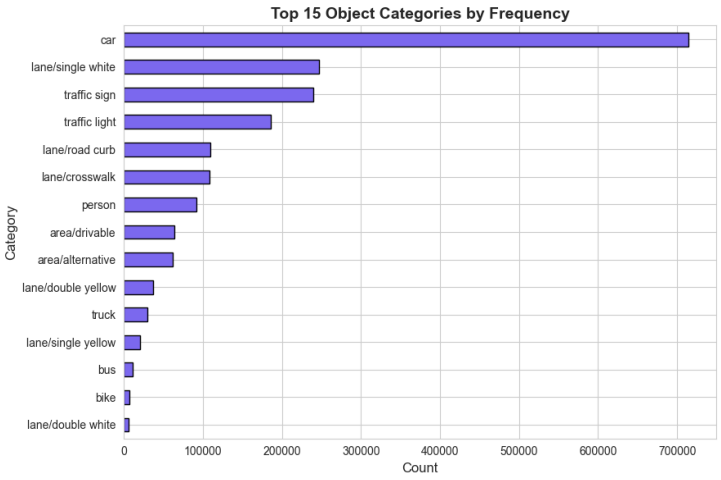
And then as mentioned in the Model exercise pdf we need to establish the baseline model using the clear-weather (over 37411 images i.e. 53.4%) images from bdd100k dataset.

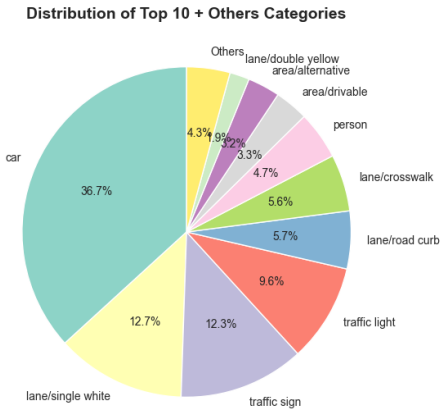
So, from the EDA I get the distribution of the weather, scene, time of the day.



Then I look at the different objects that are present in the images, so total 21 unique objects are there like car is 36.7 % and other truck, bus, person and other things are there.

```
 1. car                          : 714121 ( 36.7%)
 2. lane/single white            : 247108 ( 12.7%)
 3. traffic sign                 : 239961 ( 12.3%)
 4. traffic light                : 186301 (  9.6%)
 5. lane/road curb               : 109868 (  5.7%)
 6. lane/crosswalk               : 108284 (  5.6%)
 7. person                       : 91435 (  4.7%)
 8. area/drivable                : 64050 (  3.3%)
 9. area/alternative             : 61799 (  3.2%)
10. lane/double yellow           : 37519 (  1.9%)
11. truck                        : 30012 (  1.5%)
12. lane/single yellow           : 20220 (  1.0%)
13. bus                          : 11688 (  0.6%)
14. bike                         : 7227 (  0.4%)
15. lane/double white            : 5674 (  0.3%)
16. rider                        : 4522 (  0.2%)
17. motor                        : 3002 (  0.2%)
18. lane/single other            :  249 (  0.0%)
19. train                        :  136 (  0.0%)
20. lane/double other            :   26 (  0.0%)
21. area/unknown                 :    2 (  0.0%)
```

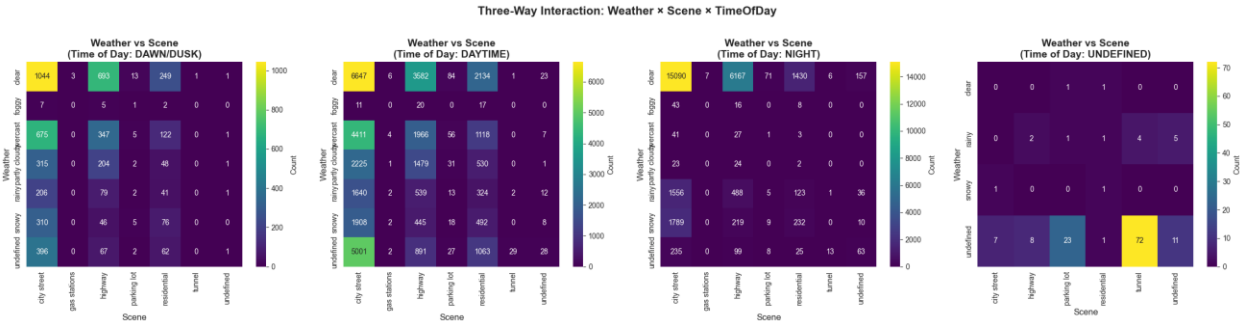


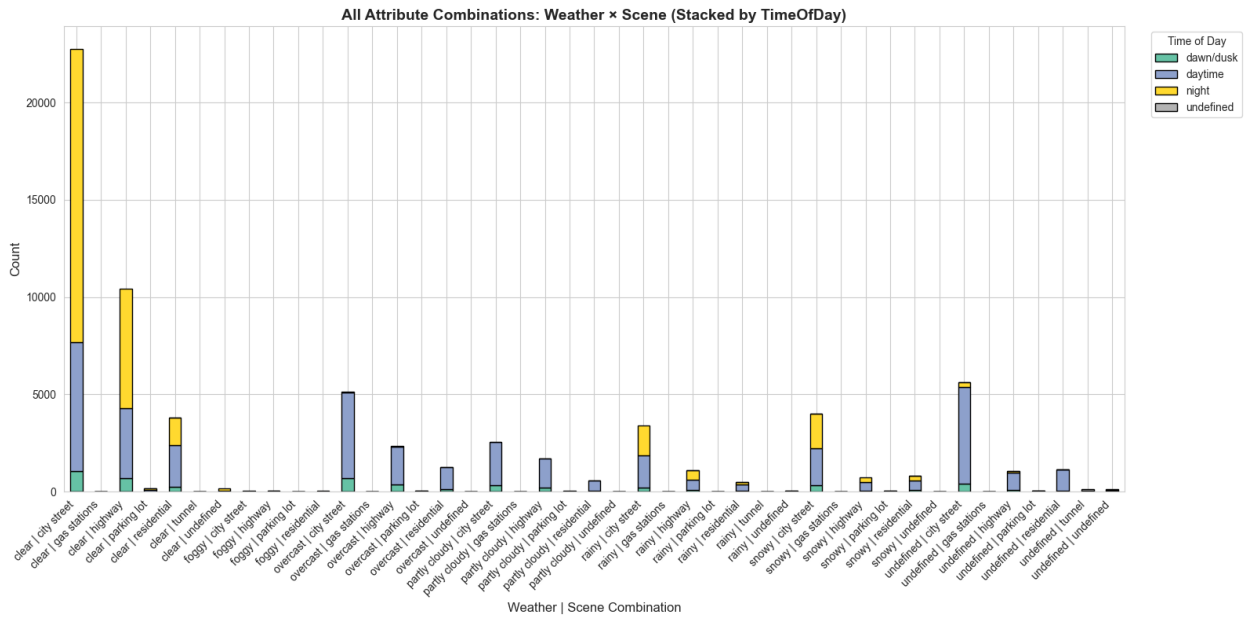


Now, I check how many samples have specific combinations like (weather: clear, scene: highway, timeofday: daytime) are present that is crucial for the data sampling so that different timeofday like dusk, daytime, night images should be present in the training/testing/validation dataset.

timeofday		dawn/dusk	daytime	night	undefined
weather	scene				
clear	city street	1044	6647	15090	0
	gas stations	3	6	7	0
	highway	693	3582	6167	0
	parking lot	13	84	71	1
	residential	249	2134	1430	1
	tunnel	1	1	6	0
	undefined	1	23	157	0

Also plotted the 3-way interaction graph between the weather, scene, time. To see how much clear weather with night images with city streets are present. And mostly data are city street, and highway, residential dataset is there.





Data Preparation

As mentioned in the model exercise to train model in the clear weather dataset I choose the 5000 images from the 37000 images out of which I included various time day, dusk, night to make sure there is variety of images while training the model.

Train 70%, validation 10%, Test 20% using the stratified sampling based on the timeofday parameter.

```
Train set size: 3500
Test set size: 1000
Val set size: 500
```

```
Selected 5000 images for the dataset.
timeofday
daytime      2500
night        2000
dawn/dusk     500
```

Now, challenge is to convert the json labels into the yolo format where class center_x, center_y, width, height in the .txt file

so data should be placed into the train, test, valid respective folders in the give format as image for yolo compatible training

```
├── test
│   ├── images
│   └── labels
├── train
│   ├── images
│   └── labels
└── valid
    ├── images
    └── labels
```

Results:

I choose the Yolo11s model which is good in accuracy and efficiency.

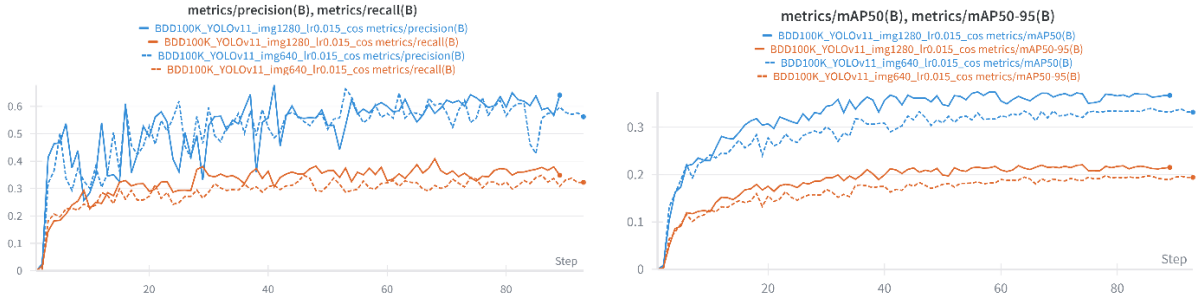
And for logging the results etc I used the Weights and bias

I choose these values

- Image size 640, 1280
- Epochs 100
- patience 15

Evaluation metrics

- mAP50 (Mean Average Precision @ IoU 0.5): Measures the model's ability to correctly classify and roughly localize objects. This is our primary metric for "detection success."
- mAP50-95: A stricter metric averaging precision over IoU thresholds from 0.5 to 0.95. This measures localization precision.



The improvement in mAP50-95 with 1280px input confirms our EDA finding regarding small object sizes. At 640px, a distant car might be reduced to a small pixel feature map, making it indistinguishable from noise. At 1280px, the model retains enough spatial resolution to resolve these features, leading to significantly better localization. Model can be train for more epochs with longer patience value with more image data

Img size 640 results true label vs predicted labels

Figure 1 left side: True label, right side: predicted labels from test set

Img 1280 results true label vs predicted labels

Figure 2 Left side: True Label, Right Side: predicted label from the test set

Look at the second image from the top and you observe the confidence of predicting the truck improve form the 640 image size to 1280 improve from 0.7 to 0.9.

but because data is highly skewed towards car classes in the future improvement we need to train it using more image data and data that is diverse including the various objects and longer training interval that might improve the model performance.