RECCOtag

March 19, 2023

1 RECCOtag.ipynb

1.1 Objective

In this experiment we measured line-of-sight detection distance for tagged CRB using a standard RECCO test tag (labelled UOG2) using 3 transceivers marked EMU1, UOG1 and UOG2. The test tag was placed on a piece of wood about 6 inches above the ground at one end of a measured linear transect. Using the most sensitive setting, the distance between the transceiver and the target was increased until no signal could be heard in the headphones. For each tranceiver, detection distance was measured three times with the test tag in two orientations: with the long axis perpendicular and parallel to the transect.

1.2 Results

- There was no significant difference in maximum detection range among the 3 devices tested (Kruskall-Wallace test; p = 0.051 for tag in perpendicular orientation, p = 0.065 for tag in parallel orientation.
- In perpendicular orientation, detection range was 183 to 225 feet.
- In parallel orientation, detection range was 38 to 78 feet.

[1]:

```
import pandas as pd
import plotly.express as px
from scipy import stats
import scikit_posthocs as sp
```

[2]:

```
df = pd.read_csv('RECCOtag.csv')
df
```

[2]:

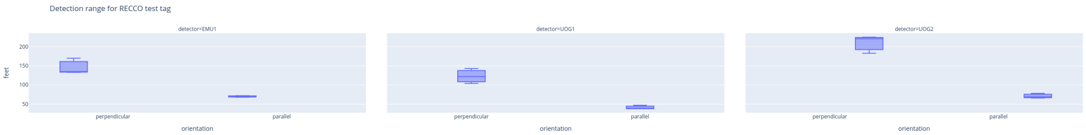
```
  detector   orientation  feet
0     EMU1 perpendicular   135
1     EMU1 perpendicular   133
2     EMU1 perpendicular   170
3     EMU1      parallel    68
4     EMU1      parallel    72
5     EMU1      parallel    70
6     UOG1 perpendicular   143
7     UOG1 perpendicular   122
8     UOG1 perpendicular   104
9     UOG1      parallel    47
```

```
10      UOG1        parallel    38
11      UOG1        parallel    38
12      UOG2   perpendicular   183
13      UOG2   perpendicular   225
14      UOG2   perpendicular   221
15      UOG2        parallel    66
16      UOG2        parallel    78
17      UOG2        parallel    69
```

```
[3]: g = df.groupby('orientation')
     g['feet'].describe()
```

```
[3]:                count        mean        std    min    25%    50%    75%    max
     orientation
     parallel         9.0   60.666667  15.337862   38.0   47.0   68.0   70.0   78.0
     perpendicular    9.0  159.555556  43.006137  104.0  133.0  143.0  183.0  225.0
```

```
[4]: px.box(data_frame=df,x='orientation',y='feet', facet_col='detector',␣
      ↪title='Detection range for RECCO test tag')
```



```
[25]: df1 = df.query("orientation=='perpendicular'")
      EMU1 = df1.query("detector=='EMU1'")['feet']
      UOG1 = df1.query("detector=='UOG1'")['feet']
      UOG2= df1.query("detector=='UOG2'")['feet']
      stats.kruskal(EMU1, UOG1, UOG2)
```

```
[25]: KruskalResult(statistic=5.955555555555556, pvalue=0.05090583233639852)
```

```
[28]: # Dunn's multiple comaparison test to find signigicant differences
      # Unnecessary in this case because the p-value from the Kruskal-Wallis test is␣
       ↪gtreater than 0.05
      sp.posthoc_dunn([EMU1, UOG1, UOG2], p_adjust = 'bonferroni')
```

```
[28]:          1         2         3
      1  1.000000  1.000000  0.303151
      2  1.000000  1.000000  0.051218
      3  0.303151  0.051218  1.000000
```

```
[29]: df1 = df.query("orientation=='parallel'")
      EMU1 = df1.query("detector=='EMU1'")['feet']
      UOG1 = df1.query("detector=='UOG1'")['feet']
      UOG2= df1.query("detector=='UOG2'")['feet']
      stats.kruskal(EMU1, UOG1, UOG2)
```

```
[29]: KruskalResult(statistic=5.46778711484594, pvalue=0.06496584833309137)
```

```
[30]: # Dunn's multiple comaparison test to find signigicant differences
      # Unnecessary in this case because the p-value from the Kruskal-Wallis test is␣
        ↪gtreater than 0.05
      sp.posthoc_dunn([EMU1, UOG1, UOG2], p_adjust = 'bonferroni')
```

```
[30]:          1         2         3
      1  1.000000  0.108314  1.000000
      2  0.108314  1.000000  0.154946
      3  1.000000  0.154946  1.000000
```

```
[ ]:
```