4.01-jk-eda-report

November 20, 2025

```
[1]: # Ensure most recent (and not cached) source code is used
     %load_ext autoreload
     %autoreload 2
```

```
[2]: # Import libraries
     import json

     import matplotlib.pyplot as plt
     #plt.style.use("seaborn-v0_8-darkgrid")
     plt.style.use("ggplot")

     plt.rcParams['text.usetex'] = True
     plt.rcParams['font.family'] = 'serif'
     %matplotlib inline

     import os

     import pandas as pd
     pd.options.display.max_columns = None
     import scipy
     import seaborn as sns
```

```
[3]: # Import functions from other python files in the project
     from pdga_score_scraper import *
     from pdga_score_scraper.config import PROCESSED_DATA_DIR
```

```
2025-11-20 11:22:32.026 | INFO     |
pdga_score_scraper.config:<module>:11 - PROJ_ROOT
```

path is: /home/jon/repos/pdga-score-scraper

```
[4]: # Try to get event_id from environment variable
     event_id = os.getenv("NB_ARGS")
     if event_id == None:
         event_id = 95560
         print(f"Environment variable not found.  Hardcoded to {event_id}.")

     else:
```

```
        print(f"Processing data for event {event_id}.")
```

Processing data for event 89433.

```
[5]:  # Import processed data
      df = pd.read_json(PROCESSED_DATA_DIR / f"{event_id}-processed.json")

      with open(PROCESSED_DATA_DIR / f"{event_id}-metadata-processed.json", 'r') as f:
          metadata = json.load(f)
```

```
[6]:  #df.head()
```

0.1 Event metadata

```
[ ]:
```

```
[7]:  ti = metadata['tournament_info']

      width = 20
      print("Event metadata\n" + 25 * '-')
      print(f"{'Name':<{width}} {ti['Name']}")
      print(f"{'Date':<{width}} {ti['DateRange']}")
      print(f"{'Location':<{width}} {ti['LocationShort']}")
      print(f"{'Total players':<{width}} {ti['TotalPlayers']}")

      divs = []
      for div in ti['Divisions']:
          divs.append(div['Division'])

      print(f"{'Divisions':<{width}} {divs}")

      l = []
      for layout in ti['Layouts']:
          l.append(layout['Name'])

      print(f"{'# of layouts':<{width}} {len(l)}")
      print(f"{'Layouts':<{width}} {l}")
```

```
Event metadata
-------------------------
Name                The Hoodoo Presented by OTB East and Discraft - MPO, FPO,
MP40, FP40, MP50, FP50, MA1
Date                Saturday, Oct 18, 2025
Location            Quakertown, PA
Total players       60
Divisions           ['MPO', 'MP40', 'MP50', 'MA1']
# of layouts        1
```

```
Layouts               ['HooDoo']
```

[8]: `#metadata`
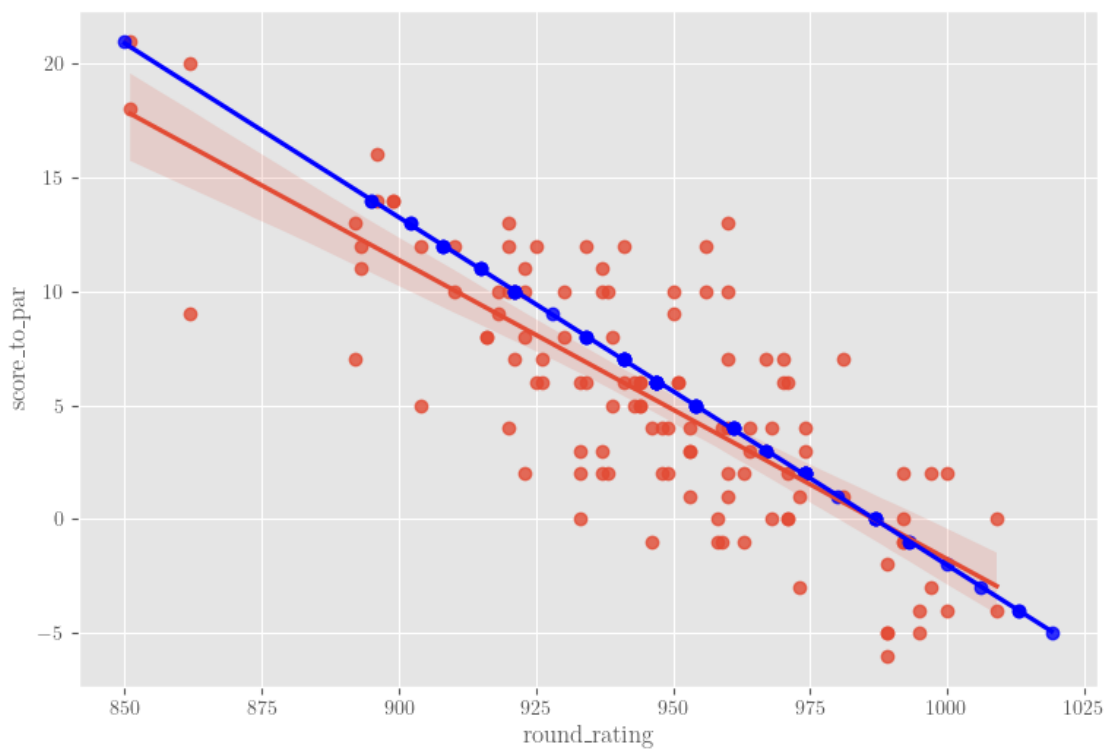
0.2 Plots

```
[9]: fig, axs = plt.subplots(1, 1, figsize = (9, 6))

     sns.regplot(data=df, x="player_rating", y = "score_to_par")
     #ax.set_title("Monkies")
     #ax.set_xlabel("PDGA player rating")
     #ax.set_ylabel("Round score relative course par")

     sns.regplot(data=df[df['round'] == 1], x="round_rating", y = "score_to_par",␣
       ↪color="blue")

     #ax.legend()

     plt.show()
```

0.3 Regressing individual hole scores against player rating

Does hole par inherently affect regresssion slope? Guessing yes. Maybe compare only equally-parred holes to evaluate differentiation.

```
[10]: slopes = {}

      fig, axs = plt.subplots(6, 3, sharey=True, sharex=True, figsize = (9, 12))
      fig.suptitle("PDGA player rating versus hole scores", fontsize=20)

      for i in range(1, 19):
          ax = axs[(i-1) // 3, (i-1)%3]

          p = sns.regplot(data=df, x="player_rating", y = f"Hole {i}", ax=ax)
          ax.set_xlabel("") # turn off to not interfere with sharex, sharey
          ax.set_ylabel("")
          ax.set_title(f"Hole {i}", fontsize=12)

          slope, intercept, r, p, sterr = scipy.stats.linregress(x=p.get_lines()[0].
       ↪get_xdata(), y=p.get_lines()[0].get_ydata())
          slopes[i] = slope

      plt.show()
```
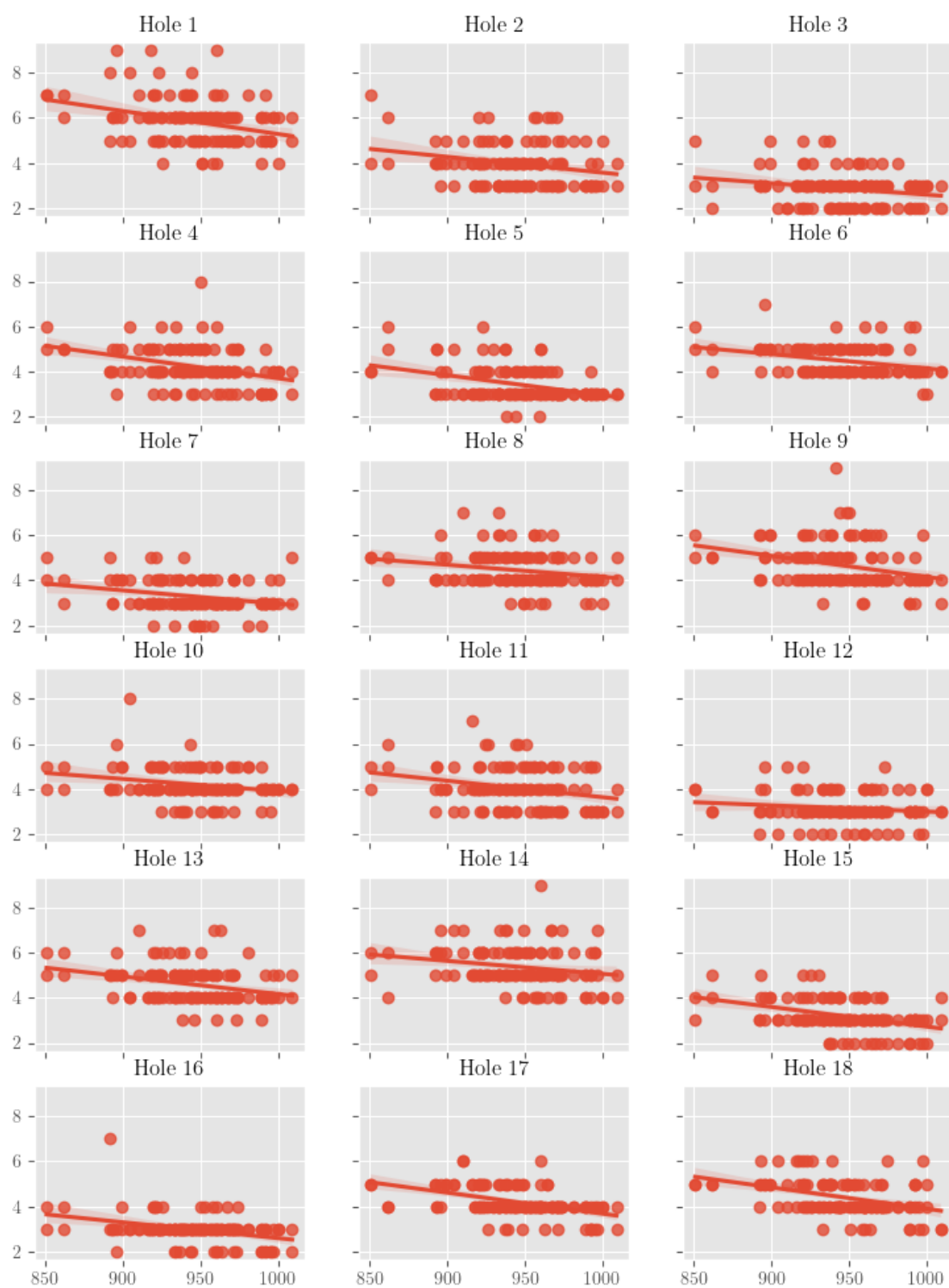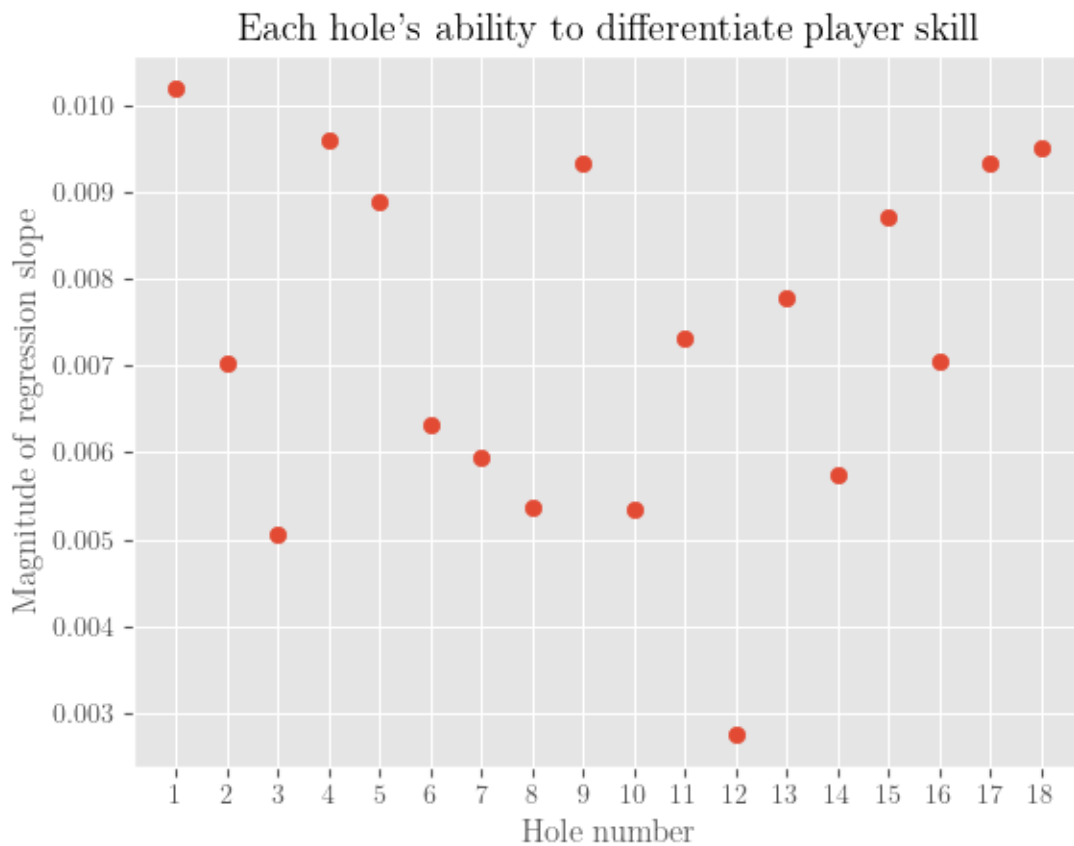
PDGA player rating versus hole scores

Hole 1 Hole 2 Hole 3
Hole 4 Hole 5 Hole 6
Hole 7 Hole 8 Hole 9
Hole 10 Hole 11 Hole 12
Hole 13 Hole 14 Hole 15
Hole 16 Hole 17 Hole 18

```
[11]: plt.scatter(slopes.keys(), [-1 * x for x in slopes.values()])
      plt.xticks(range(1, 19))
      plt.xlabel("Hole number")
      plt.ylabel("Magnitude of regression slope")
      plt.title("Each hole's ability to differentiate player skill")
      plt.show()
```



```
[12]: s = {k: round(v, 4) for k, v in sorted(slopes.items(), key=lambda item:
      ↪item[1])}
      s.keys()
```

```
[12]: dict_keys([1, 4, 18, 17, 9, 5, 15, 13, 11, 16, 2, 6, 7, 14, 8, 10, 3, 12])
```

0.4 Metadata

```
[13]: metadata['tournament_info']
```

```
[13]:  {'DateRange': 'Saturday, Oct 18, 2025',
        'EndDate': '2025-10-18',
        'Country': 'United States',
        'Location': 'Quakertown, Pennsylvania',
        'LocationShort': 'Quakertown, PA',
        'Divisions': [{'DivisionID': 100,
          'Division': 'MPO',
          'DivisionName': 'Mixed Pro Open',
          'Players': 31,
          'IsPro': True,
          'ShortName': 'Mixed Pro Open',
          'AbbreviatedName': 'Mixed Pro Open',
          'LatestRound': 2},
         {'DivisionID': 111,
          'Division': 'MP40',
          'DivisionName': 'Mixed Pro 40+',
          'Players': 4,
          'IsPro': True,
          'ShortName': 'Mixed Pro 40+',
          'AbbreviatedName': 'Mixed Pro 40+',
          'LatestRound': 2},
         {'DivisionID': 121,
          'Division': 'MP50',
          'DivisionName': 'Mixed Pro 50+',
          'Players': 5,
          'IsPro': True,
          'ShortName': 'Mixed Pro 50+',
          'AbbreviatedName': 'Mixed Pro 50+',
          'LatestRound': 2},
         {'DivisionID': 400,
          'Division': 'MA1',
          'DivisionName': 'Mixed Amateur 1',
          'Players': 20,
          'IsPro': False,
          'ShortName': 'Mixed Amateur 1',
          'AbbreviatedName': 'Mixed Am 1',
          'LatestRound': 2}],
        'Finals': 'no',
        'HighestCompletedRound': 2,
        'LatestRound': 2,
        'FinalRound': 2,
        'Layouts': [{'LayoutID': 721578,
          'CourseID': 25812,
          'CourseName': 'Nockamixon State Park',
          'Name': 'HooDoo',
          'Holes': 18,
          'Par': 68,
```

```
    'Length': 9289,
    'Units': 'Feet',
    'Accuracy': 'M',
    'Details': [{'Hole': 'H1', 'Label': '1', 'Par': 5, 'Length': 845},
     {'Hole': 'H2', 'Label': '2', 'Par': 4, 'Length': 447},
     {'Hole': 'H3', 'Label': '3', 'Par': 3, 'Length': 220},
     {'Hole': 'H4', 'Label': '4', 'Par': 4, 'Length': 644},
     {'Hole': 'H5', 'Label': '5', 'Par': 3, 'Length': 304},
     {'Hole': 'H6', 'Label': '6', 'Par': 4, 'Length': 626},
     {'Hole': 'H7', 'Label': '7', 'Par': 3, 'Length': 390},
     {'Hole': 'H8', 'Label': '8', 'Par': 4, 'Length': 570},
     {'Hole': 'H9', 'Label': '9', 'Par': 4, 'Length': 541},
     {'Hole': 'H10', 'Label': '10', 'Par': 4, 'Length': 565},
     {'Hole': 'H11', 'Label': '11', 'Par': 4, 'Length': 404},
     {'Hole': 'H12', 'Label': '12', 'Par': 3, 'Length': 317},
     {'Hole': 'H13', 'Label': '13', 'Par': 4, 'Length': 617},
     {'Hole': 'H14', 'Label': '14', 'Par': 5, 'Length': 769},
     {'Hole': 'H15', 'Label': '15', 'Par': 3, 'Length': 358},
     {'Hole': 'H16', 'Label': '16', 'Par': 3, 'Length': 377},
     {'Hole': 'H17', 'Label': '17', 'Par': 4, 'Length': 542},
     {'Hole': 'H18', 'Label': '18', 'Par': 4, 'Length': 753}]}],
 'Name': 'The Hoodoo Presented by OTB East and Discraft - MPO, FPO, MP40, FP40,
MP50, FP50, MA1',
 'NameHtml': 'The Hoodoo Presented by OTB East and Discraft - MPO, FPO, MP40,
FP40, MP50, FP50, MA1',
 'MultiLineName': {'pre': '',
  'main': 'The Hoodoo',
  'post': 'Presented by OTB East and Discraft - MPO, FPO, MP40, FP40, MP50,
FP50, MA1'},
 'MultiLineNameHtml': ' <span class="event-name-main">The Hoodoo</span> <span
class="event-name-post">Presented by OTB East and Discraft - MPO, FPO, MP40,
FP40, MP50, FP50, MA1</span>',
 'RawTier': 'B',
 'Rounds': 2,
 'RoundsList': {'1': {'Number': 1,
   'Label': 'Round 1',
   'LabelAbbreviated': 'Rd 1',
   'Date': '2025-10-18',
   'DateAbbreviated': 'Oct 18',
   'ShowDate': False},
  '2': {'Number': 2,
   'Label': 'Round 2',
   'LabelAbbreviated': 'Rd 2',
   'Date': '2025-10-18',
   'DateAbbreviated': 'Oct 18',
   'ShowDate': False}},
 'Semis': 'no',
```

'SimpleName': 'The Hoodoo',
'StartDate': '2025-10-18',
'TDName': 'Steve Braud',
'TDPDGANum': 16027,
'Tier': 'B',
'FormattedTier': 'B-Tier',
'FormattedLongTier': 'B-Tier',
'TierPro': 'B',
'TierAm': 'B',
'TierX': False,
'TotalPlayers': 60,
'TournamentId': '89433',
'International': False,
'FullStats': True,
'TimeZone': 'America/New_York',
'AdditionalEventInfo': None,
'EventVideo': None,
'ScoringFormat': 'S'}

[ ]: