HUMAN RESOURCE ANALYSIS USING PYTHON

by

2010030151 K S Satyavarsan

2010030071 Jyothin Movva

2010030040 Devaraj Acharya

2010030341 Snehith Kamepally

2010030329 Karthik

ABSTRACT

* The field of human resources analysis, which can be understood as an approach to human recourses management focused on data and analytical thinking.
* Human resources analysis an optimal use of human resources to ensure that the human resources of an organization remain an asset and not a liability.
* The project analyzes the data of the employees working in the organization.

LITERATURE REVIEW

1 . **Human Resource Analysis with Python (**AMAN KHARWAL**):**

**Abstract-**The field of human resources analysis, which can be understood as an approach to human resources management focused on data and analytical thinking, is quickly becoming an indispensable part of organizational configurations.

**2. Social Network Analysis in Human Resource Development: A New Methodology (**JOHN-PAUL HATALA, Louisiana State University**):**

**Abstract-**Through an exhaustive review of the literature, this article looks at the applicability of social network analysis (SNA) in the field of human resource development

**3. Utility Analysis for Decisions in Human Resource Management (**John W. Boudreau Cornell University**):**

**Abstract-**This chapter will discuss utility analysis (VA), which attempts to answer such questions by focusing on decisions about human resources. Utility analysis refers to the process that describes, predicts and/or explains what determines the usefulness or desirability of decision options, and examines how that information affects decisions

METHODS

* The project is coded in python
* The dependencies that are used are
  + Numpy – mathematical functions
  + Pandas – data analysis and manipulation
  + Matplotlib – plotting library
  + Seaborn – visualization library based on matplotlib
  + Ploty – plot 2d data and color scaling functionality

IMPLEMENTATION FLOWCHART

Diagram

Description automatically generated

FUNCTIONING OF THE CODE

Libraries:

**from** flask **import** Flask, render\_template

**from** flask\_assets **import** Bundle, Environment

**import** pandas **as** pd

**import** json

**import** plotly

**import** plotly.express **as** px

**import** csv

Flask initialization

app **=** Flask(\_\_name\_\_)

assets **=** Environment(app)

**if** \_\_name\_\_ **==** "\_\_main\_\_":

    app.run(*debug***=**True)

Generating new column from the data using conditions

*def* train\_data():

    df **=** pd.read\_csv("aug\_train.csv")

**with** open('aug\_train.csv',*newline***=**'') **as** file:

        data **=** csv.DictReader(file)

        count **=** 0

**for** row **in** data:

**if** row['education\_level'] **==** 'Graduate' **or** row['education\_level'] **==** 'Phd':

**if** *int*(row['training\_hours']) **>=** 10:

                    df.loc[count, 'hit\_counter'] **=** 'hit'

**else**:

                df.loc[count, 'hit\_counter'] **=** 'not\_hit'

            print(count)

            count**+=**1

        df.to\_csv("aug\_train.csv", *index***=**True)

flask links and chart plotting

@app.route('/')

*def* index():

**return** render\_template('index.html')

@app.route('/about')

*def* about():

**return** render\_template('about.html')

@app.route('/tra1n')

*def* tra1n():

    train\_data()

**return** render\_template('tra1n.html',*description***=**"Data is trained")

@app.route('/chart1')

*def* chart1():

    fig **=** px.imshow(train.isnull().T)

    fig.update\_layout(*title***=**'Missing values in data set')

    #fig.show()

    graphJSON **=** json.dumps(fig, *cls***=**plotly.utils.PlotlyJSONEncoder)

    header**=**"chart1"

    description **=** "temp1"

**return** render\_template('notdash2.html', *graphJSON***=**graphJSON, *header***=**header,*description***=**description)

@app.route('/chart2')

*def* chart2():

    plot\_city **=** train['city'].value\_counts()[0:50].reset\_index()

    plot\_city.columns **=** ['City','Count']

    px.bar(plot\_city,*x***=**'City',*y***=**'Count',*title***=**'City',*color***=**'Count')

    fig **=** px.bar(plot\_city,*x***=**'City',*y***=**'Count',*template***=**'gridon',*title***=**'City',*color***=**'Count')

    fig.update\_layout(*title\_text***=**'plot\_city',*title\_y***=**0.5)

    graphJSON **=** json.dumps(fig, *cls***=**plotly.utils.PlotlyJSONEncoder)

    header**=**"chart2"

    description **=** "temp2"

**return** render\_template('notdash2.html', *graphJSON***=**graphJSON, *header***=**header,*description***=**description)

@app.route('/chart3')

*def* chart3():

    plot\_cdi **=**train['city\_development\_index'].value\_counts().reset\_index()[0:50]

    plot\_cdi.columns **=** ['cdi','Count']

    plot\_cdi['cdi'] **=** plot\_cdi['cdi'].astype('str')

    fig **=** px.bar(plot\_cdi,*y***=**"Count", *x***=**"cdi",*color***=**'Count',*title***=**'City development index')

    graphJSON **=** json.dumps(fig, *cls***=**plotly.utils.PlotlyJSONEncoder)

    header**=**"chart3"

    description **=** "temp3"

**return** render\_template('notdash2.html', *graphJSON***=**graphJSON, *header***=**header,*description***=**description)

@app.route('/chart4')

*def* chart4():

    plot\_gender **=** train['enrolled\_university'].value\_counts().reset\_index()

    plot\_gender.columns **=** ['enrolled\_university','count']

    fig **=** px.pie(plot\_gender,*values***=**'count',*names***=**'enrolled\_university',*title***=**'enrolled\_university')

    graphJSON **=** json.dumps(fig, *cls***=**plotly.utils.PlotlyJSONEncoder)

    header**=**"chart4"

    description **=** "temp4"

**return** render\_template('notdash2.html', *graphJSON***=**graphJSON, *header***=**header,*description***=**description)

@app.route('/chart5')

*def* chart5():

    plot\_gender **=** train['education\_level'].value\_counts().reset\_index()

    plot\_gender.columns **=** ['education\_level','count']

    fig **=** px.pie(plot\_gender,*values***=**'count',*names***=**'education\_level',*title***=**'education\_level')

    graphJSON **=** json.dumps(fig, *cls***=**plotly.utils.PlotlyJSONEncoder)

    header**=**"chart5"

    description **=** "temp5"

**return** render\_template('notdash2.html', *graphJSON***=**graphJSON, *header***=**header,*description***=**description)

@app.route('/chart6')

*def* chart6():

    plot\_gender **=** train['major\_discipline'].value\_counts().reset\_index()

    plot\_gender.columns **=** ['major\_discipline','count']

    fig **=** px.pie(plot\_gender,*values***=**'count',*names***=**'major\_discipline',*title***=**'Major discipline')

    fig.update\_traces(*textposition***=**'inside', *textinfo***=**'percent+label')

    graphJSON **=** json.dumps(fig, *cls***=**plotly.utils.PlotlyJSONEncoder)

    header**=**"chart6"

    description **=** "temp6"

**return** render\_template('notdash2.html', *graphJSON***=**graphJSON, *header***=**header,*description***=**description)

@app.route('/chart7')

*def* chart7():

    plot\_gender **=** train['company\_size'].value\_counts().reset\_index()

    plot\_gender.columns **=** ['company\_size','count']

    fig **=** px.pie(plot\_gender,*values***=**'count',*names***=**'company\_size',*title***=**'company\_size is determined by no. of people employees')

    graphJSON **=** json.dumps(fig, *cls***=**plotly.utils.PlotlyJSONEncoder)

    header**=**"chart7"

    description **=** "temp7"

**return** render\_template('notdash2.html', *graphJSON***=**graphJSON, *header***=**header,*description***=**description)

@app.route('/chart8')

*def* chart8():

    train **=** pd.read\_csv('aug\_train.csv')

    plot\_gender **=** train['hit\_counter'].value\_counts().reset\_index()

    plot\_gender.columns **=** ['hit\_counter','count']

    fig **=** px.pie(plot\_gender,*values***=**'count',*names***=**'hit\_counter',*title***=**'desired employment graph')

    graphJSON **=** json.dumps(fig, *cls***=**plotly.utils.PlotlyJSONEncoder)

    header**=**"chart8"

    description **=** "temp8"

**return** render\_template('notdash2.html', *graphJSON***=**graphJSON, *header***=**header,*description***=**description)

HTML FILES

Index.html

  <!DOCTYPE html>

<html lang="en">

<head>

  <meta charset="utf-8">

  <meta content="width=device-width, initial-scale=1.0" name="viewport">

  <title>HOME</title>

  <meta content="" name="description">

  <meta content="" name="keywords">

  <!-- Google Fonts -->

  <link href="https://fonts.googleapis.com/css?family=Open+Sans:300,300i,400,400i,600,600i,700,700i|Roboto:300,300i,400,400i,500,500i,700,700i&display=swap" rel="stylesheet">

  <!-- Vendor CSS Files -->

  <link href = "{{ url\_for('static',filename='css/style.css')}}" rel = "stylesheet">

  <link href = "{{ url\_for('static',filename='vendor/animate.css/animate.min.css')}}" rel = "stylesheet">

  <link href = "{{ url\_for('static',filename='vendor/aos/aos.css')}}" rel = "stylesheet">

  <link href = "{{ url\_for('static',filename='/vendor/bootstrap/css/bootstrap.min.css')}}" rel = "stylesheet">

  <link href = "{{ url\_for('static',filename='vendor/bootstrap-icons/bootstrap-icons.css')}}" rel = "stylesheet">

  <link href = "{{ url\_for('static',filename='vendor/boxicons/css/boxicons.min.css')}}" rel = "stylesheet">

  <link href = "{{ url\_for('static',filename='vendor/glightbox/css/glightbox.min.css')}}" rel = "stylesheet">

  <link href = "{{ url\_for('static',filename='vendor/swiper/swiper-bundle.min.css')}}" rel = "stylesheet">

  <!-- =======================================================

  \* Template Name: Moderna - v4.8.0

  \* Template URL: https://bootstrapmade.com/free-bootstrap-template-corporate-moderna/

  \* Author: BootstrapMade.com

  \* License: https://bootstrapmade.com/license/

  ======================================================== -->

</head>

<body>

  <!-- ======= Header ======= -->

  <header id="header" class="fixed-top d-flex align-items-center header-transparent">

    <div class="container d-flex justify-content-between align-items-center">

      <div class="logo">

        <h1 class="text-light"><a href="index.html"><span>HUMAN RESOURCE MANAGEMENT</span></a></h1>

        <!-- Uncomment below if you prefer to use an image logo -->

        <!-- <a href="index.html"><img src="assets/img/logo.png" alt="" class="img-fluid"></a>-->

      </div>

      <nav id="navbar" class="navbar">

        <ul>

          <li><a class="active " href="/">Home</a></li>

          <li><a href="about">About</a></li>

          <li><a href="tra1n">Train</a></li>

          <li class="dropdown"><a href="#"><span>Graphs</span> <i class="bi bi-chevron-down"></i></a>

            <ul>

              <!--<li><a href="chart1">missing values</a></li>-->

              <li><a href="chart2">City Distribution</a></li>

              <li><a href="chart3">City Development Index</a></li>

              <li><a href="chart4">Enrolled Univertisty Index</a></li>

              <li><a href="chart5">Educational Level Index</a></li>

              <li><a href="chart6">Major Descipline</a></li>

              <li><a href="chart7">Company Size</a></li>

              <li><a href="chart8">Desired Employment Chart</a></li>

              </ul>

          </li>

        </ul>

        <i class="bi bi-list mobile-nav-toggle"></i>

      </nav><!-- .navbar -->

    </div>

  </header><!-- End Header -->

  <!-- ======= Hero Section ======= -->

  <section id="hero" class="d-flex justify-cntent-center align-items-center">

    <div id="heroCarousel" class="container carousel carousel-fade" data-bs-ride="carousel" data-bs-interval="5000">

      <!-- Slide 1 -->

      <div class="carousel-item active">

        <div class="carousel-container">

          <h2 class="animate\_\_animated animate\_\_fadeInDown">WELCOME TO HUMAN-RESOUCE MANAGEMENT</h2>

          <p class="animate\_\_animated animate\_\_fadeInUp">Use the drop down option to browse through the graphs</p>

        </div>

      </div>

   </div>

  </body>

</html>

Plotty.html

<!**doctype** html>

<**html**>

    <**head**>

        <**meta** charset="utf-8">

        <**meta** content="width=device-width, initial-scale=1.0" name="viewport">

        <**title**>chart</**title**>

        <**meta** content="" name="description">

        <**meta** content="" name="keywords">

        <!-- Google Fonts -->

        <**link** href="https://fonts.googleapis.com/css?family=Open+Sans:300,300i,400,400i,600,600i,700,700i|Roboto:300,300i,400,400i,500,500i,700,700i&display=swap" rel="stylesheet">

        <!-- Vendor CSS Files -->

        <**link** href = "{{ url\_for('static',filename='css/style.css')}}" rel = "stylesheet">

        <**link** href = "{{ url\_for('static',filename='vendor/animate.css/animate.min.css')}}" rel = "stylesheet">

        <**link** href = "{{ url\_for('static',filename='vendor/aos/aos.css')}}" rel = "stylesheet">

        <**link** href = "{{ url\_for('static',filename='/vendor/bootstrap/css/bootstrap.min.css')}}" rel = "stylesheet">

        <**link** href = "{{ url\_for('static',filename='vendor/bootstrap-icons/bootstrap-icons.css')}}" rel = "stylesheet">

        <**link** href = "{{ url\_for('static',filename='vendor/boxicons/css/boxicons.min.css')}}" rel = "stylesheet">

        <**link** href = "{{ url\_for('static',filename='vendor/glightbox/css/glightbox.min.css')}}" rel = "stylesheet">

        <**link** href = "{{ url\_for('static',filename='vendor/swiper/swiper-bundle.min.css')}}" rel = "stylesheet">

        <!-- =======================================================

        \* Template Name: Moderna - v4.8.0

        \* Template URL: https://bootstrapmade.com/free-bootstrap-template-corporate-moderna/

        \* Author: BootstrapMade.com

        \* License: https://bootstrapmade.com/license/

        ======================================================== -->

      </**head**>

      <**header** id="header" class="fixed-top  align-items-center header-opaque ">

        <**div** class="container d-flex justify-content-between align-items-center">

          <**div** class="logo">

            <**h1** class="text-light"><**a** href="index.html"><**span**>HUMAN RESOURCE MANAGEMENT</**span**></**a**></**h1**>

          </**div**>

          <**nav** id="navbar" class="navbar">

            <**ul**>

              <**li**><**a** class="active " href="/">Home</**a**></**li**>

              <**li**><**a** href="about">About</**a**></**li**>

              <**li**><**a** href="tra1n">Train</**a**></**li**>

              <**li** class="dropdown"><**a** href="#"><**span**>Graphs</**span**> <**i** class="bi bi-chevron-down"></**i**></**a**>

                <**ul**>

                  <!--<li><a href="chart1">missing values</a></li>-->

                  <**li**><**a** href="chart2">City Distribution</**a**></**li**>

                  <**li**><**a** href="chart3">City Development Index</**a**></**li**>

                  <**li**><**a** href="chart4">Enrolled Univertisty Index</**a**></**li**>

                  <**li**><**a** href="chart5">Educational Level Index</**a**></**li**>

                  <**li**><**a** href="chart6">Major Descipline</**a**></**li**>

                  <**li**><**a** href="chart7">Company Size</**a**></**li**>

                  <**li**><**a** href="chart8">Desired Employment Chart</**a**></**li**>

                  </**ul**>

              </**li**>

            </**ul**>

            <**i** class="bi bi-list mobile-nav-toggle"></**i**>

          </**nav**><!-- .navbar -->

        </**div**>

      </**header**><!-- End Header -->

  <**div** id="chart" class="chart"></**div**>

</**body**>

<**script** id="graph" src="https://cdn.plot.ly/plotly-latest.min.js"></**script**>

<**script** id = "graph" type="text/javascript">

*var* graphs **=** {{graphJSON | safe}};

    Plotly.plot('chart',graphs,{});

</**script**>

</**html**>

GITHUB COMMITS

Graphical user interface, text, chat or text message

Description automatically generated

Text

Description automatically generated

APLHA TESTING

Graphical user interface, application

Description automatically generated

BETA TESTING

A computer screen capture

Description automatically generated with low confidence

Text

Description automatically generated

A computer screen capture

Description automatically generated with medium confidence

Graphical user interface, text, application

Description automatically generated

Graphical user interface, application, table, timeline

Description automatically generated

Graphical user interface

Description automatically generated