

SAS Visual Analytics Data Driven Contents(DDC) Guide

SAS Visual Analytics Data Driven Contents(DDC) Guide

차트 구성 방법

RADAR 차트

DATA Set 업로드

VA 서버에 radar.html 업로드 하기

신규 리포트에 DDC 컴포넌트 추가 및 설정

결과 화면

멀티 Y-축차트

DATA Set 업로드

VA 서버에 multi.html 업로드 하기

신규 리포트에 DDC 컴포넌트 추가 및 설정

결과 화면

차트 구성 방법

SAS Visual Analytics 에서 제공하는 Data Driven Contents(DDC)는 외부 3rd Party 라이브러리를 참조하여 VA 에서 제공하지 않는 리포트 오브젝트를 만드는데 사용 합니다.

DDC 기능을 구현하는데 있는 크게 2가지 방법이 있습니다.

1. VA 서버내에 웹페이지를 Deployment 하여 구성하는 방법
2. VA 서버 외부의 웹서버에 웹페이지를 Deplymnet 하여 구성하는 방법

외부 사이트의 웹사이트 내에 웹페이지와 연동시 반듯이 SSL(HTTPS) 통신으로만 연동이 됩니다.
따라서 외부 사이트가 SSL(HTTPS) 를 지원하는 웹서버 혹은 WAS 서버 여야만 합니다.

본 가이드 에서는 **Chart.js** 의 RADAR 차트와 **Plot.ly** 의 multy Y 축 차트를 만드는 예제로 구성되어 있습니다.

RADAR 차트

DATA Set 업로드

다운로드(Dropbox) : [다운로드 클릭](#)

csv 파일을 다운로드 하여 CAS 테이블 생성

VA 서버에 radar.html 업로드 하기

SAS VA 서버의 /opt/sas/viya/home/var/www/html/htmlcommons/ 디렉토이에 업로드

radar.html

```
<HTML>
<head>
  <script src="https://cdnjs.cloudflare.com/ajax/libs/Chart.js/2.7.2/Chart.min.js">
</script>
</head>
<style>
  canvas {
    -moz-user-select: none;
    -webkit-user-select: none;
    -ms-user-select: none;
  }

  #chartjs-tooltip {
    opacity: 1;
    position: absolute;
    background: rgba(0, 0, 0, .7);
    color: white;
    border-radius: 3px;
    -webkit-transition: all .1s ease;
    transition: all .1s ease;
    pointer-events: none;
    -webkit-transform: translate(-50%, 0);
    transform: translate(-50%, 0);
  }

  #chartjs-radar {
    width: 60%;
    height: 60%;
  }

  .chartjs-tooltip-key {
    display: inline-block;
    width: 10px;
    height: 10px;
    margin-right: 10px;
  }
</style>

<BODY>
  <div id="chartjs-radar">
    <canvas id="myChart"></canvas>
  </div>
</BODY>
<script>
  if (window.addEventListener) {
    // For standards-compliant web browsers
```

```

        window.addEventListener("message", onMessage, false);
    } else {
        window.attachEvent("onmessage", onMessage);
    }
    function onMessage(event) {
        console.log(event.data);
        var color = Chart.helpers.color;
        window.chartColors = {
            orange: 'rgb(255, 159, 64)',
            yellow: 'rgb(255, 205, 86)',
            green: 'rgb(75, 192, 192)',
            blue: 'rgb(54, 162, 235)',
            purple: 'rgb(153, 102, 255)',
            grey: 'rgb(231,233,237)',
            red: 'rgb(255, 99, 132)',
        };

        data1 = event.data.data[0];
        data2 = event.data.data[1];

        var ctx = document.getElementById("myChart");

        var data = {
            labels: ["Cycling", "Swimming", "Climbing", "Shooting", "Running"],
            datasets: [
                {
                    label: "Peter",
                    fillColor: "rgba(220,220,220,0.5)",
                    strokeColor: "rgba(220,220,220,1)",
                    data: data1,
                    backgroundColor:
color(window.chartColors.red).alpha(0.2).rgbString(),
                    borderColor: window.chartColors.red,
                    pointBackgroundColor: window.chartColors.red
                },
                {
                    label: "Jane",
                    fillColor: "rgba(151,187,205,0.5)",
                    strokeColor: "rgba(151,187,205,1)",
                    data: data2,
                    backgroundColor:
color(window.chartColors.blue).alpha(0.2).rgbString(),
                    borderColor: window.chartColors.blue,
                    pointBackgroundColor: window.chartColors.blue
                }
            ]
        };

        var myRadarChart = new Chart(ctx, {
            type: 'radar',
            data: data,
        });
    }

```

```

}
</script>
</BODY>
</HTML>

```

신규 리포트에 DDC 컴포넌트 추가 및 설정

1. Object Pane 에서 Other > Data-Driven Content 를 선택하여 Canvas 에 Drag&Drop 합니다.

The screenshot shows the SAS Visual Analytics interface. On the left, the 'Object' pane is open, showing a list of components. Under the 'Other' category, 'Data-Driven Content' is selected. The main canvas displays a table with the following data:

	Happiness Rank	Happiness Score	Economy (GDP per Capita)	Family	Health (Life Expectancy)	Freedom	Trust (Government Corruption)	Generosity	Dystopia Residual
	1	7.526	1.44178	1.16374	0.79504	0.57941	0.44453	0.36171	2.73939
	2	7.509	1.52733	1.14524	0.86303	0.58557	0.41203	0.28083	2.69463
	3	7.501	1.42666	1.18326	0.86733	0.56624	0.14975	0.47678	2.83137
	4	7.498	1.57744	1.1269	0.79579	0.59609	0.35776	0.37895	2.66465
	5	7.413	1.40598	1.13464	0.81091	0.57104	0.41004	0.25492	2.82596
	6	7.404	1.44015	1.0961	0.8276	0.5737	0.31329	0.44834	2.70485
	7	7.339	1.46488	1.02912	0.81231	0.55211	0.29927	0.47416	2.70749
aland	8	7.334	1.36066	1.17278	0.83096	0.58147	0.41904	0.49401	2.47553
aland	9	7.313	1.44443	1.10476	0.8512	0.56837	0.32331	0.47407	2.5465
	10					0.58218	0.40867	0.38254	2.54734
thern	11	7.267	1.33766	0.99537	0.84917	0.36432	0.08728	0.32288	3.31029
	12	7.119	1.45038	1.08383	0.80565	0.54355	0.21348	0.32865	2.69343
	13	7.104	1.50796	1.04782	0.779	0.48163	0.14868	0.41077	2.72782
	14	7.087	1.06879	1.02152	0.76146	0.55225	0.10547	0.22553	3.35168
	15	7.039	1.35943	1.08113	0.77758	0.46823	0.12275	0.22202	3.0076

2. 오른쪽 Options Pane 에서 Web Content > URL 부분에 `http://{va-server-ip}/htmlcommons/radar.html` 을 입력합니다.

The screenshot shows the 'Options' pane for the 'Data-Driven Content' component. The 'Web Content' section is expanded, and the 'URL' field is set to `http://10.96.4.98/htmlcommons/radar.html`.

3. 오른쪽 Roles Pane 에서 **Variables** 에 아이템을 추가 합니다.

아이템 추가 반듯이 unique 값인 id 컬럼도 추가 해야함

Search [?] sasdemo

Add Data Items

- ☒ id - 4
- ☒ A
- ☒ B
- ☒ C
- ☒ D
- ☒ E
- ☐ Frequency
- ☐ Frequency Percent

OK Cancel

	Freedom	Trust (Government Corruption)	Gen
04	0.57941	0.44453	
03	0.58557	0.41203	
33	0.56624	0.14975	
79	0.59609	0.35776	
91	0.57104	0.41004	
76	0.5737	0.31329	
31	0.55211	0.29927	
96	0.58147	0.41904	
12	0.56837	0.32331	0.47407 2.5465

Data Roles

Data-Driven Content 1

Variables*

+ Add

Options

Roles

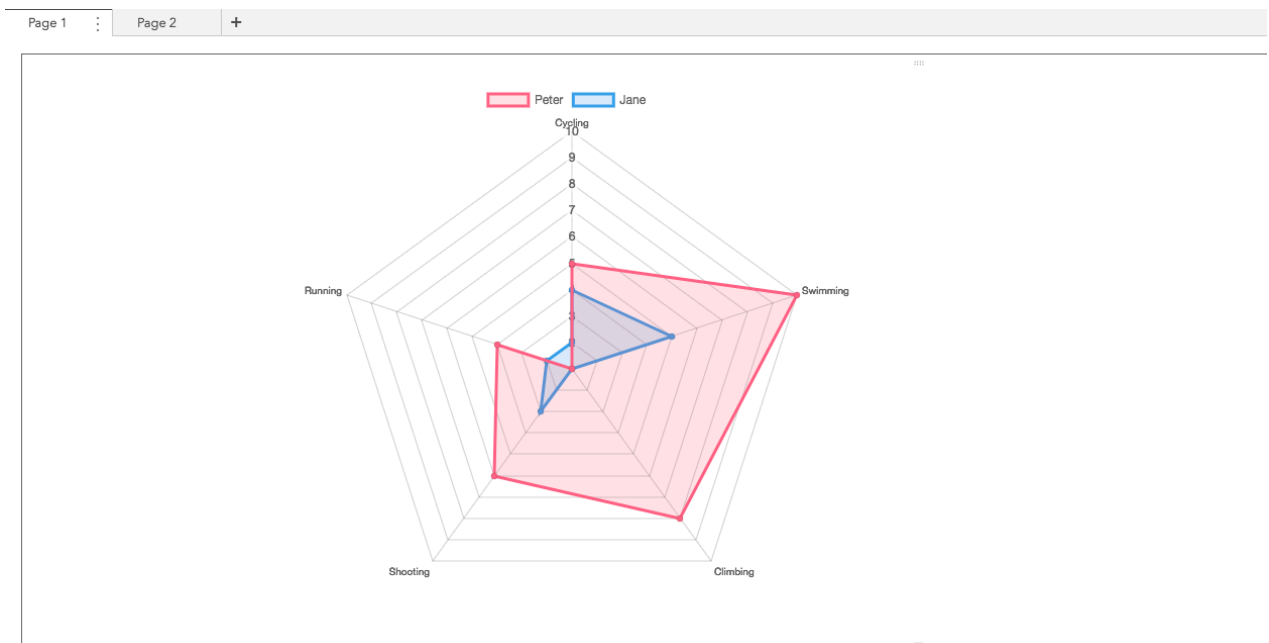
Actions

Rules

Filters

Ranks

결과 화면



멀티 Y-축트차트

DATA Set 업로드

다운로드(Dropbox) : [다운로드 클릭](#)

csv 파일을 다운로드 하여 CAS 테이블 생성

VA 서버에 multi.html 업로드 하기

```
<HTML>

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

<BODY>
  <div id="myDiv" style="width:100%;height:100%;"></div>
</BODY>
<script>
  if (window.addEventListener) {
    // For standards-compliant web browsers
    window.addEventListener("message", onMessage, false);
  } else {
    window.attachEvent("onmessage", onMessage);
  }
  // Retrieve data and begin processing
  function onMessage(event) {
    if (event && event.data) {
      rowCount = event.data.rowCount;
      var base_name = [];
      var sensor1 = [];
      var sensor2 = [];
      var sensor3 = [];
      var sensor4 = [];
      for (i = 0; i < rowCount; i++) {
        base_name[i] = event.data.data[i][0];
      }

      for (i = 0; i < rowCount; i++) {
        sensor1[i] = event.data.data[i][1];
      }
      var trace1 = {
        x: base_name,
        y: sensor1,
        name: 'Sensor1',
        type: 'scatter'
      };

      for (i = 0; i < rowCount; i++) {
        sensor2[i] = event.data.data[i][2];
      }
      var trace2 = {
        x: base_name,
        y: sensor2,
        name: 'Sensor2',
        yaxis: 'y2',
        type: 'scatter'
      };
    }
  }
}
```

```

for (i = 0; i < rowCount; i++) {
    sensor3[i] = event.data.data[i][3];
}
var trace3 = {
    x: base_name,
    y: sensor3,
    name: 'Sensor3',
    yaxis: 'y3',
    type: 'scatter'
};

for (i = 0; i < rowCount; i++) {
    sensor4[i] = event.data.data[i][4];
}
var trace4 = {
    x: base_name,
    y: sensor4,
    name: 'Sensor4',
    yaxis: 'y4',
    type: 'scatter'
};

var data = [trace1, trace2, trace3, trace4];

var layout = {
    title: 'Multiple Y-Axes Example',
    yaxis: {
        title: 'Sensor1',
        titlefont: { color: '#1f77b4' },
        tickfont: { color: '#1f77b4' }
    },
    yaxis2: {
        title: 'Sensor2',
        titlefont: { color: '#ff7f0e' },
        tickfont: { color: '#ff7f0e' },
        anchor: 'free',
        overlaying: 'y',
        side: 'left',
        position: 0.15
    },
    yaxis3: {
        title: 'Sensor3',
        titlefont: { color: '#d62728' },
        tickfont: { color: '#d62728' },
        anchor: 'x',
        overlaying: 'y',
        side: 'right'
    },
    yaxis4: {
        title: 'Sensor4',
        titlefont: { color: '#9467bd' },
        tickfont: { color: '#9467bd' },

```

```

        anchor: 'free',
        overlaying: 'y',
        side: 'right',
        position: 0.85
    }
};

Plotly.newPlot('myDiv', data, layout);

var getUrlParameter = function getUrlParameter(sParam) {
    var sPageURL =
decodeURIComponent(window.location.search.substring(1)),
        sURLVariables = sPageURL.split('&'),
        sParameterName,
        i;

    for (i = 0; i < sURLVariables.length; i++) {
        sParameterName = sURLVariables[i].split('=');

        if (sParameterName[0] === sParam) {
            return sParameterName[1] === undefined ? true :
sParameterName[1];
        }
    }
};

var tech = getUrlParameter('parameter');
console.log(tech);
}
}
</script>
</HTML>

```

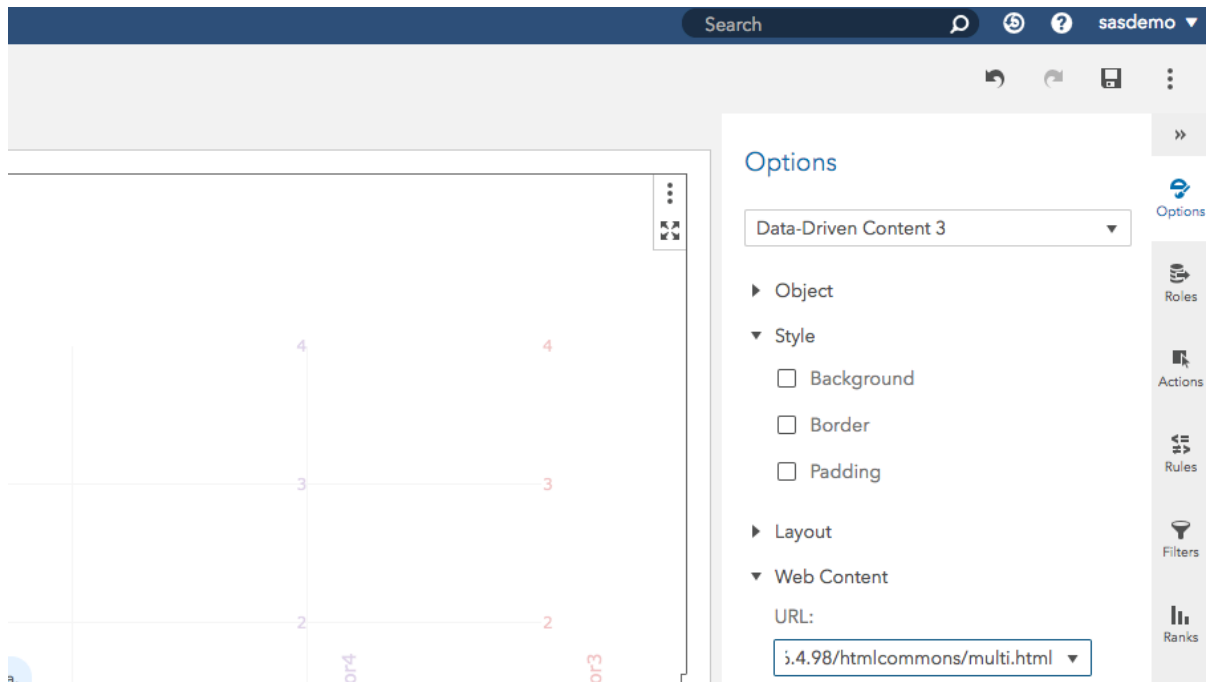
신규 리포트에 DDC 컴포넌트 추가 및 설정

1. Object Pane 에서 Other > Data-Driven Content 를 선택하여 Canvas 에 Drag&Drop 합니다.

The screenshot shows the SAS Visual Analytics interface. On the left, the 'Objects' pane is open, showing a list of components. Under the 'Other' category, 'Data-Driven Content' is selected. The main canvas displays a table of data with columns: Happiness Rank, Happiness Score, Economy (GDP per Capita), Family, Health (Life Expectancy), Freedom, Trust (Government Corruption), Generosity, and Dystopia Residual. The 'Data Roles' pane on the right shows 'Data-Driven Content 1' selected. A tooltip message states: 'No data items are assigned. Assign one or more columns to view the data.'

Happiness Rank	Happiness Score	Economy (GDP per Capita)	Family	Health (Life Expectancy)	Freedom	Trust (Government Corruption)	Generosity	Dystopia Residual
1	7.526	1.44178	1.16374	0.79504	0.57941	0.44453	0.36171	2.73939
2	7.509	1.52733	1.14524	0.86303	0.58557	0.41203	0.28083	2.69463
3	7.501	1.42666	1.18326	0.86733	0.56624	0.14976	0.47678	2.83137
4	7.498	1.57744	1.1269	0.79579	0.59609	0.35776	0.37895	2.66465
5	7.413	1.40598	1.13464	0.81091	0.57104	0.41004	0.25492	2.82596
6	7.404	1.44015	1.0961	0.8276	0.5737	0.31329	0.44834	2.70485
7	7.339	1.46468	1.02912	0.81231	0.55211	0.29927	0.47416	2.70749
8	7.334	1.36066	1.17278	0.83096	0.58147	0.41904	0.49401	2.47553
9	7.313	1.44443	1.10476	0.8512	0.56837	0.32331	0.47407	2.5466
10					0.58218	0.40867	0.38254	2.54734
11	7.267	1.33766	0.99537	0.84917	0.36432	0.08728	0.32288	3.31029
12	7.119	1.45038	1.08383	0.80565	0.54355	0.21348	0.32865	2.69343
13	7.104	1.50796	1.04782	0.779	0.48163	0.14868	0.41077	2.72782
14	7.087	1.06879	1.02152	0.76146	0.55225	0.10547	0.22553	3.35168
15	7.039	1.35943	1.08113	0.77758	0.46823	0.12275	0.22202	3.0076

- 오른쪽 Options Pane 에서 Web Content > URL 부분에
http://{va-server-ip}/htmlcommons/multi.html 을 입력합니다.



- 오른쪽 Roles Pane 에서 **Variables** 에 아이템을 추가 합니다.

time, sensor1, sensor2, sensor3, sensor4 순서를 지켜 입력해야함

결과 화면

