

# Table des matières

## Sommaire

Table des matières .....	1
Projet l'existant.....	2
But :.....	2
Aperçu .....	2
Amélioration de projet existant .....	2
Tâches faites .....	2
nom tâche : affichages de tous les axes et grilles par rapport à la carte : .....	2
But et méthodes employées : .....	2
Capture d'écran.....	4
nom tâche : affichage des infos sur les points .....	4
But et méthodes employées : .....	4
Capture d'écran.....	4
nom tâche : Zoom .....	5
But et méthodes employées : .....	5
Factorisation de code .....	6

# Projet l'existant

## But :

Le but de ce projet était de visualiser de données (ces données sont générées d'une manière aléatoire via le programme R ) d'une manière interactive sur les cartes auto- organisatrices de Kohonen.

L'interactivité est faite via la bibliothèque D3.js. Les cartes interactives existantes, ses données étaient représentées sous forme d'un histogramme.

Notre but était entre autres de transformer les histogrammes en nuage de points et d'établir l'interactivité sur ces points.

## Aperçu

A mettre capture d'écran de l'histogramme

## Amélioration de projet existant

### Tâches faites

**nom tache : affichages de tous les axes et grilles par rapport à la carte :**

**But et méthodes employées :**

**But :**

Le but était d'afficher les axes et grilles sur chacune des hexagones de la carte.

**Méthodes employées :**

Code	Commentaires
<pre> for (var indice = 0; indice &lt; nbRows*nbColumns; indice++) {     var g=d3.select("#gcircle" +(indice+1));      g.append("g")     .attr("class", "x axis") // permet de tracer      .call(xAxe)     .attr("transform", "translate("+ 0 +","+ (width/2 +10) +")" );      g.append("g")     .attr("class", "y axis")// permet de tracer      .call(yAxe);  } </pre>	<p>On parcourt toutes les colonnes et lignes de la carte Avec la boucle for.</p> <p>On sélectionne la grille où mettre les axes via</p> <pre> var g=d3.select("#gcircle" +(indice+1)); </pre> <p>et on dessine les axes et grilles via les instructions suivantes :</p> <pre> g.append("g") .attr("class", "x axis") .call(xAxe) .attr("transform", "translate("+ 0 +","+ (width/2 +10) +")" );  g.append("g") .attr("class", "y axis")// permet de tracer .call(yAxe); </pre> <p>La méthode call(xAxe) fait appelle à ceci :</p> <pre> var xAxe = d3.svg.axis()     .scale(échelleX)     .tickSize(1)     .ticks(4)     .orient("bottom") </pre> <p>Ci haut scale fixe l'échelle de l'axe et orient le sens ou l'orientation de l'axe.</p> <p>Pour call(yAxe), pareil on fait appel à var yAxe cette fois ci :</p> <pre> var yAxe = d3.svg.axis()     .scale(échelleY)     .tickSize(1)     .ticks(4)     .orient("left") </pre>

--	--

## Capture d'écran

**nom tâche : affichage des infos sur les points**

**But et méthodes employées :**

### *But :*

Le but est qu'on survole de la souris sur les points de pouvoir afficher les coordonnées de points.

### *Méthodes employées :*

Code	Commentaire
<pre>function affichelabel(label,x,y){     var ch=label + ':' + x + '+' + y;     d3.select("#grid-ref").text(ch); }</pre>	<p>La création de la fonction qui prend en argument le label venant de la table nuageData.label[indice] , nuageData.x[indice] (pour l'abscisse), nuageData.y[indice] (pour l'ordonnée)</p>
<pre>for (var indice=0; indice&lt;nuageData.cell.length; indice++){</pre>	<p>On parcourt chaque cellule de l'hexagone</p>
<pre>var g=d3.select("#gcircle" + nuageData.cell[indice]);</pre>	<p>On sélectionne le cellule approprié.</p>
<pre>g.append("circle") .attr("onmouseover","affichelabel(" + nuageData.label[indice] + ',' + nuageData.x[indice] + ',' + nuageData.y[indice] + ");") .attr("cx",function(){</pre>	<p>On ajoute les cercle s et l'événement onmouseover Tout en appelant la fonction affichelabel</p>

## Capture d'écran

**nom tache : Zoom**

**But et méthodes employées :**

**But :**

Au clic de la souris sur une de cellules, de pouvoir supprimer la partie non nécessaire au zoom tout en décalant les axes.

**Méthodes employées**

Code	Commentaire
<pre>var g=d3.select("#gcircle" +d3.select(this).attr("hexa"));</pre>	On sélectionne la cellule de l'hexagone
<pre>g.select(".x.axis").remove();</pre>	On supprime l'ancien axe
<pre>var echelleX = d3.scale.linear() .domain([d3.select(this).attr("x")- 1,d3.max(nuageData.x)]) .range([0,width/2 +10])</pre>	On crée le nouvel échelle
<pre>var xAxe = d3.svg.axis() .scale(echelleX) .tickSize(1) .ticks(4) .orient("bottom")</pre>	On récrée l'axe avec le nouvel échelle
<pre>g.selectAll("circle").each( function(){ if(d3.select(this).attr("x")&lt;x) d3.select(this).remove();</pre>	On supprime les points pas importants au zoom  On selectionne tous les cercles et on les supprime par rapport au point cliqué.

<pre>}  );</pre>	
--------------------------	--

*Capture d'écran*

## Factorisation de code