Investing in what works for better lives

Social Investment Data Foundation Code User Guide

Guidelines for creating a dataset for social investment analysis*

Version 1: December 2017

SIA-2017-0250

Creative Commons Licence

Liability

While all care and diligence has been used in processing, analysing and extracting data and information in this publication, the Social Investment Agency gives no warranty it is error free and will not be liable for any loss or damage suffered by the use directly, or indirectly, of the information in this publication.

Citation

Social Investment Agency (2017). *Social Investment Data Foundation Code User Guide*. Wellington, New Zealand.

ISBN 978-0-9951022-0-0 (online)

Published in December 2017 and scheduled for review in April 2019
Social Investment Agency
Wellington, New Zealand

Intended audience

This guide is written for analysts and policy people to support data analytics. However we have adopted a plain English style of language to make the guide as accessible as possible.

Resources, tools and guides

The SIA is developing a range of tools, products and guidance to enable agencies to develop their social investment approaches, and analyse and measure the impact and effectiveness of the services they're delivering.

** By 'social investment analysis' we mean analysis of anonymised, integrated data with the objective of measuring the impact of government services on people's lives.*

Analysing data for social investment

Analytic methods are vital to the social investment approach. We need to be able to see the effectiveness of social services across the full spectrum of outcomes known to the social sector, and to be able to test whether or not the initiatives we deliver are benefiting the New Zealanders who are receiving them.

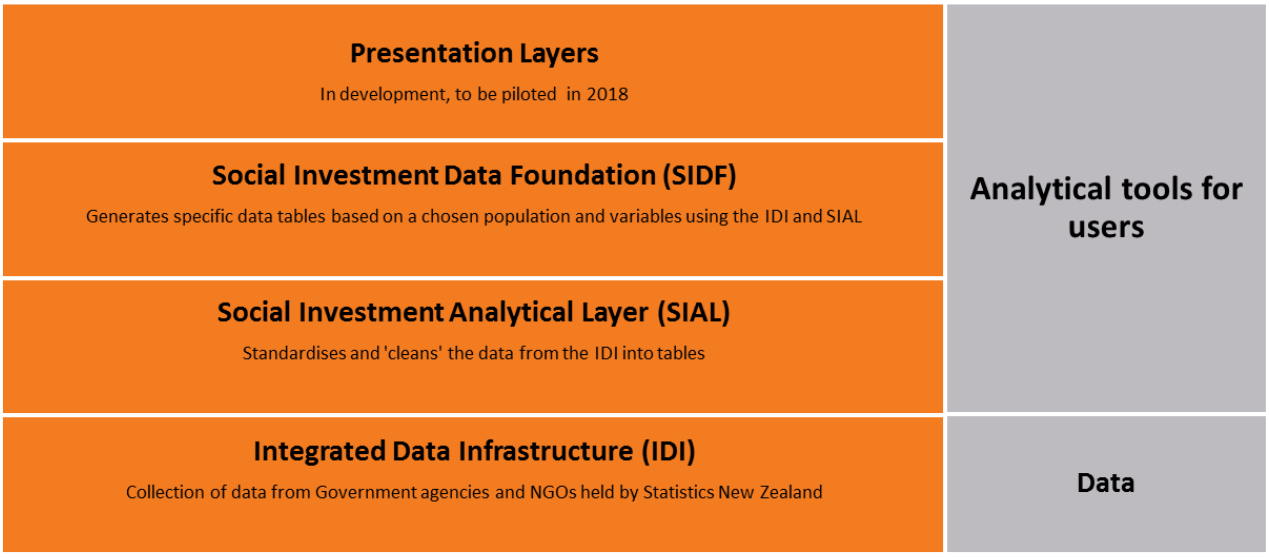
Statistics New Zealand provides a world-leading resource for social investment analytics: the Integrated Data Infrastructure (IDI). It is a research database containing multiple datasets collected by government agencies and non-governmental organisations.

Using this integrated data resource, analysts are able to generate insights which help us understand what works, for whom, at what cost.

Analytical tools to simplify analysis

The datasets available in the IDI are not structured in a standard accessible way. To make it easier for analysts to make use of this data, the SIA has created tools which simplify its use. The focus of this guide is the Social Investment Data Foundation (SIDF). The SIDF is a piece of code which generates usable datasets from the IDI according to the user's specifications.

The SIDF was developed to be used in conjunction with a suite of analytical tools produced by the SIA:



Presentation Layers

In development, to be piloted in 2018

Social Investment Data Foundation (SIDF)

Generates specific data tables based on a chosen population and variables using the IDI and SIAL

Social Investment Analytical Layer (SIAL)

Standardises and 'cleans' the data from the IDI into tables

Integrated Data Infrastructure (IDI)

Collection of data from Government agencies and NGOs held by Statistics New Zealand

Analytical tools for users

Data

Before the SIDF code can be applied, the data needs to be structured with the Social Investment Analytical Layer (SIAL). The SIDF builds on the SIAL code to produce a dataset that is ready for analysis. The SIAL user guide is available on the SIA website. We recommend that all analysts working with IDI data make use of these tools to save time and effort, and avoid duplication of work.

This guide explains how to apply the SIDF code to enable analysis of IDI data. Although the SIDF code is publically available, you will need access to a Statistics New Zealand data lab to apply it to IDI data. You can learn more about how to apply for IDI access in the SIA's Beginner's Guide to the IDI.

The Social Investment Data Foundation

The SIDF generates a dataset for a population of interest, and applies other variables to be analysed.

Structuring the data with the SIAL

The SIAL code formats and structures the IDI datasets, and documents business rules to reformat data from the IDI into events-based tables. This format encourages consistency in the definitions of variables, reduces analysis time, and enables greater understanding of the data. Each responsible agency provides quality assurance of their data and business rules.

The tables generated by the SIAL code are a required input for the SIDF. They are used to create service metric variables at the individual level.

Datasets ready for analysis

The SIDF code builds on the SIAL code and produces a dataset that is ready for analysis. Different projects require different datasets, so the SIDF code has the flexibility to tailor pre-set variables to the user's specifications. Users can specify who, when and what they are interested in analysing, and a dataset ready for their analysis is then created.

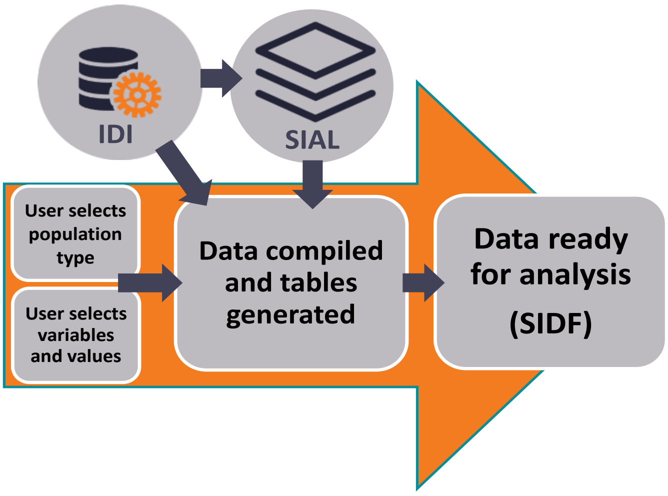
Analysis variables in the dataset are based on the user's population dataset, a chosen 'as at' date, and the time periods (e.g. months) required before and after this date. The code has default values, meaning users do not need to make specifications for all variables.

It is also possible for advanced users to make adjustments beyond the default settings.

SIDF code is modular – allowing for greater flexibility

Being modular means the code is split into separate programs. This makes it easy for users to make changes to the program if required, as changes to individual modules will not affect other parts of the program and individual components can be run as standalone modules. It also means a user can more easily understand the program and its individual parts.

Figure 1: Summary of the SIDF code and input:


IDI
SIAL
User selects population type
User selects variables and values
Data compiled and tables generated
Data ready for analysis (SIDF)

Who is in the SIDF dataset?

There are three kinds of population that the SIDF code will handle as an input – snapshot, cohort and bespoke populations.

Snapshot population

A snapshot population is the entire resident population at a specified point in time. For example: every New Zealander in New Zealand on 1 January 2000. This is referred to as an estimated resident population (ERP) and is linked to the IDI spine.

Cohort population

A cohort population is a group linked by a common date. For example: all New Zealanders who were 50-years-old in 2000 or all New Zealanders who applied for a benefit in 2005. It is also an ERP that is linked to the IDI spine.

Bespoke population

A bespoke population is a list of individuals defined using any other business rules. For example: a list of individuals who received a service. This method also requires a specified date. A user can either specify a point in time date (as in the snapshot method), or a defined period of time (as in the cohort method).

What variables are generated by the SIDF dataset code?

A dataset generated using the SIDF will always have unique identifiers and 'as at' dates. In addition to these, users can specify the variables they are interested in generating.

There are three different types of variables a user can specify:

- characteristic variables
- service metric variables
- outcome variables.

Characteristic variables

Characteristic variables capture information about an individual that either does not change, or is slow changing.

The characteristics variables are listed in Table 1. By default all characteristics variables listed in Table 1 are available for users to specify which ones they want to use to generate a dataset.

Table 1: Full list of characteristics variables generated by the SIDF code

Variable name	Description
Age	An individual’s age, derived from their date of birth. Age is generated based on the specified ‘as at’ date.
Agency flags	A flag that identifies if ‘identification’ was captured in an agency’s collection (e.g. Ministry of Education indicator, Ministry of Social Development (MSD) indicator, etc.). Binary 1/0 indicator.
Birth month	An individual’s birth month.
Birth year	An individual’s birth year
Date of birth	Derived using the birth month and year. Day of birth set as 15.
Ethnicity (prioritised)	Ethnicity prioritised as per Statistics New Zealand business rules.
Gender	An individual’s gender. Gender is generated on the specified ‘as at’ date.
Iwi	An individual’s Iwi. Iwi is generated as at the 2013 census, not based on the ‘as at’ date specified by the user. This is because it is the only data source that captures Iwi for the whole population in the IDI. Up to three Iwi chosen.
Person indicator	Indicator of ‘identification’ belonging to a person (and not a business ‘identification’).
Region	The region an individual lives in. Region is generated on the specified ‘as at’ date.
Spine indicator	Indicator of ‘identification’ being part of the IDI spine.
TA code	The territorial authority an individual lives in. Generated as on the specified ‘as at’ date.

Service metric variables

Service metrics are available through the SIDF. They capture information about an event, aggregated at the individual level, during a specified period of time. These metrics include, but are not limited to:

- total cost of an event (cst)
- total duration of an event (dur)
- number of times an event has occurred (cnt)
- count from start date (ct2)
- days since first (dffe)
- days since last (dsle).

Table 2 provides an example of a service event table. Table 3 provides an example of how this table is aggregated to create a service metric for an individual.

Table 2: Example of a service event table using synthetic data

snz_uid	department	datamart	subject_area	start_date	end_date	event_type	cost
1234	MSD	BEN	T1	01/01/2013	31/12/2014	JSS	$10,987.65
9876	MSD	BEN	T1	03/03/2014	12/12/2014	YPP	$1,234.56
1234	MSD	BEN	T1	30/06/2015	30/09/2015	JSS	$1,234.56

Table 3: Example of aggregated service metrics using synthetic data

snz_uid	f_msd_ben_t1_cnt	f_msd_ben_t1_dur	f_msd_ben_t1_cst
1234	2	793	$12,222.21
9876	1	284	$1,234.56

The naming convention for service metric variables is as follows:

$$<W>[int]\_<DEP>\_<DTM>\_<SUB>[\_EV1|\ldots|\_EVn]\_<VAR>$$

- W = the window. This can be either a time period prior to the specified ‘as at’ date (the profile window, *p*) or after the specified ‘as at’ date (the forecast window, *f*).
- int = an integer which denotes the period number.
- DEP = the department where the data comes from, e.g. MSD, MOH, etc. When data has come from combined set of departments the agency MIX is used.
- DTM = the data mart (e.g. BEN for benefits).
- SUB = subject area (e.g. T1 for tier one main benefits).
- EV1…EV*n* = event_type to event_type_*n.*
- VAR = the variable type (e.g. count (CNT)).

Table 4: Example of service metric variables

Variable name	Description
P1_MSD_BEN_T1_675_CNT	Count of MSD Tier 1 Job Seeker benefit events in the first period of the profile window.
P_MSD_BEN_T1_CST	Total cost of MSD Tier 1 benefit costs in the profile window.

Outcome variables

Outcome variables capture information about an individual that changes, usually as a result of receiving a service. The distinction between these and characteristics is small, but they are created in a distinct module due to the (often) complex way the code needs to be written.

The only outcome variable that the SIDF code currently generates is highest qualification (Highest_qual) – an individual's highest educational qualification on the 'as at' date. The code allows for outcome variable business rules to be easily added. It is anticipated that, over time, more outcome variables will be added to the SIDF code by future users applying the code to their own projects. Examples of future outcome variables are: a NEET (not in education, employment or training) indicator and a Before School check score variable.

Accessing and using the SIDF code

The SIDF code is hosted on GitHub, a dedicated online hosting service where the open source community can share code. It can be viewed and downloaded from:

https://github.com/nz-social-investment-agency/social_investment_data_foundation.

Before you begin using the SIDF you will need access to an IDI data lab; there are instructions for this process in the Beginner's Guide to the IDI.

Installation

1. Ensure you have an IDI project and access to the datasets you need so you can run the code.
2. Confirm you have the SIAL tables in your schema. If not, you will have to download the social investment analytical layer zip file from GitHub and follow the installation instructions in that repository first.
3. Download the zipped file for the social investment data foundation from GitHub.
4. Email the zipped file(s) to access2microdata@stats.govt.nz and ask them to move it into your project folder.
5. Unzip the files into your project.

Instructions for building the SIDF

The scripts you need to modify are contained in the **sasprogs** folder. There are three scripts in here you will run:

1. sasprogs/si_get_cohort.sas reads in your population.
2. sasprogs/si_control.sas is where you specify the arguments used to build the data foundation.
3. sasprogs/si_main.sas is where you create all of the variables. There are many of them so they are created in separate tables depending on their type.

You can join the tables together as you please to create the data foundation – an analysis-ready dataset. Due to the large number of combinations possible, it has been left to the user to choose how they build their final dataset. An example of how one was built can be found in examples/si_join_tables_manual_example_pd.sas.

STEP A Create population

1. Start a new SAS session
2. Open sasprogs/si_get_cohort.sas.
3. Populate with the code necessary to build your population. The inputs will vary depending on what populations you are interested in but the final output should be a table that has a set of ‘identifications’ and a date for each. The date is the reference date for the variables to be created ‘as at’ and needs to be in a datetime format.
4. Run the code and make sure you are happy with the output that has been produced (a table that has a set of ‘identifications’ and a date – in datetime format – for each). Refer to the example folder for an example.
5. When you are happy, make note of: the table name, the ‘identification’ column name and the date column name.

STEP B Specify arguments

1. Open **sasprogs\si_control.sas**. This is where you specify the arguments needed to build the SIDF and what variables you want to generate. Read the header to understand all the variables.

1. Scroll down to the yellow datalines and specify your arguments after the comma for each parameter. Do not put spaces before or after your arguments and make sure all of the arguments have a value, don't leave them blank. If you have trouble, remember that the arguments are referred to in the header or check **examples/si_control_example_pd.sas** for an example.
2. Make sure the values for **si_pop_table_out**, **si_id_col** and **si_asat_date** match what you made note of in step A5.
3. Run this script so SAS can identify the variables you require. It will generate a wide dataset and also put all your arguments into global macro variables.

STEP C Generate variables

1. Now that you have specified a population (step A) and some arguments needed for the SIDF (step B) you are ready to run the main scripts. Open **sasprogs/si_main.sas**.
2. Scroll down to the first **%let** statement. This is where you specify where you put the data foundation root folder. You will need to change this to reflect where you put the SIDF files.
3. Save **sasprogs/si_main.sas**.

4. Scroll down a few lines to a set of **%include** statements. Notice that they refer to the two files that you created in steps 1 and 2. Unless you've changed the names of these files (which you don't need to do) you don't need to make changes here.

5. Run **sasprogs/si_main.sas**. If your population is < 100,000 (and doesn't require main benefits or pharmaceutical data) it should only take a few minutes. The larger the target population, the longer it will take.

Look at the results

Open up the work library and explore the tables. The script **sasprogs/si_main.sas** explains what some of the tables represent. Refer to the documentation in the documents folder for descriptions of each variable. Note you might receive the warning – **"WARNING: Amount type is NA, so any Price Index/Discounting adjustments will be forced to NA".** This is OK and means that when there are no available costs, they cannot be inflated, deflated, or discounted. If you don't receive this warning that's OK too, it means all the costs you've asked for are available, or you've not asked for costs. After looking at the results you will notice multiple tables were created with many variables. Usually you would want these all in one dataset so you can begin your analysis. The next steps help you do this.

STEP D Create your Social Investment Data Foundation (an analysis ready dataset)

1. Open **examples/si_join_tables_manual_example_pd.sas**. This shows you a manual method of selecting a subset of the variables to be joined into a single table using a data step and hash objects. You can modify the data step to make sure you are selecting the variables you want. In future releases it is intended that an automated version will be made available. The automated method will return all available variables for your population. Depending on your arguments in step B this can be over 1,000 variables. Generally, you only want to choose this method if you have an easy way to choose the variables you want to analyse (over 1000 is too many). For example, variable clustering or correlation testing you can apply to a whole dataset (with categorical and numeric variables).
1. Use **examples/si_join_tables_manual_example_pd.sas** to give you an idea of how to write a script to manually join the tables. Save this script in **sasprogs**.
2. Run the script that you just created and confirm that it runs without error.
3. You are finished. Your Social Investment Data Foundation (an analysis-ready dataset) is ready to use.

Example code

An example of how to run the code end-to-end is available in the README of the GitHub repository .

This example involves running the SIDF with the (approximately) top 10,000 SNZ_UIDs from the personal details table based on the first day of each person's birthday month in 2014. It is approximately 10,000 because we confirm they are attached to the IDI spine and that they have a birth month.

It is strongly recommended that first time users run the example to become familiar with the SIDF framework first.

Advanced users

Advanced users who are interested in adding additional variables to the code are able to do so. The code is written in a way that helps you write your additions in an automated way. You can change the scripts in the **sasautos** folder if you want to do this. The scripts ending in _**ext** are the scripts you can add to.

Additional characteristic variables can be added into the **sasautos/si_get_characteristics_ext.sas** script. Additional outcomes variables can be added to the **sasautos/si_get_outcomes_ext.sas** script.